Ethics Inf Technol (2014) 16:1–13
DOI 10.1007/s10676-013-9328-z

ORIGINAL PAPER

Lockbox: mobility, privacy and values in cloud storage

Luke Stark · Matt Tierney

Published online: 2 October 2013

Abstract This paper examines one particular problem of values in cloud computing: how individuals can take advantage of the cloud to store data without compromising their privacy and autonomy. Through the creation of Lockbox, an encrypted cloud storage application, we explore how designers can use reflection in designing for human values to maintain both privacy and usability in the cloud.

Keywords Privacy · Cloud computing · Human–computer interaction (HCI) · Values and design · Usability · Cryptography · Mobility · User empowerment · Autonomy · Reflective design

Introduction

Sociologist Zygmunt Bauman has coined the term "liquid modernity" to describe the fluidity and uncertainty of contemporary sociotechnical affairs (Bauman 2007). Yet in computational terms, "liquid modernity" is already one phase transition behind the technological times: computer scientists and scholars of digital media, information and the law are increasingly focused on the capacities, both potential and already realized, of online distributed systems popularly known by the term "cloud computing."

Cloud computing enables "ubiquitous, convenient, on-demand network access" to a shared collection of computer systems, which might include "networks, servers, storage, applications, and [other] services" (Mell and Grance 2011). Access to shared information stored in the cloud is enabled by an "abstracted [and] virtualized" software interface, often via a web browser, and connected to a remote infrastructure: an individual's user data is stored alongside the data of others on servers potentially many kilometers from the user's point of access (Bauman 2007; Hon et al. 2011). Cloud computing is therefore characterized by the following features: "(i) on-demand, self-serve access; (ii) broad network access that agnostically accepts all devices—including computers, laptops, smart phones, game consoles and other network-enabled devises; (iii) resource pooling; (iv) rapid elasticity or scalability; and (v) measured service optimization" (Garon 2011; Mell and Grance 2011). In other words, cloud computing purportedly makes storing and sharing digital data cheap and convenient for both the companies that provide services in the cloud and their users. Companies offering cloud storage services benefit from inexpensive bulk data storage provided by third party-operated "server farms," such as those supporting Amazon Web Service's Simple Storage Service (S3); centralized evaluation tools able to assess usage at a granular level; and the ability to move and manage data within and between servers while providing a consistent interface for individuals accessing their data.

Cloud storage services have received considerable attention in recent years from both academics and the public despite their relatively short history as widely available commercial products. This attention has often

L. Stark (✉)
Department of Media, Culture, and Communication,
New York University, 239 Greene Street, 8th Floor,
New York, NY 10003, USA
e-mail: luke.stark@nyu.edu

M. Tierney
Department of Computer Science, Courant Institute
of Mathematical Sciences, New York University,
251 Mercer Street,, New York, NY 10012, USA
e-mail: tierney@cs.nyu.edu

been focused on the benefits of cloud computing in general, and cloud storage in particular, in terms of convenience and affordability; however, scholars and others have also begun to voice concerns about cloud computing, on technical and broader social and ethical grounds. Market leaders in cloud storage, file synching, sharing, and versioning—such as Dropbox and Google Docs—have come under scrutiny. Concerns include the contents of the terms of service (ToS) of cloud storage companies, particularly in relation to privacy (Gain 2011; Soghoian 2011); observations of the "expansive" nature of the copyright claims that have been brought to bear by cloud storage services on stored user data (Constantine 2012); and the legal and technical complexities of cloud storage business models (for instance, Dropbox's use of Amazon S3 as its storage provider arguably renders Dropbox's terms of use moot from the perspective of law enforcement officials interested in gaining access to user data (Governor 2011)).

A broader set of technical and ethical issue loom over these aforementioned concerns: the privacy of data stored in the cloud and its vulnerability to privacy breaches, unauthorized or unanticipated access, and circulation outside the control of individual users and even the service providers themselves (Matthews 2011; Pearson et al. 2009). Nonetheless, supporting privacy in cloud storage is under-theorized from an explicitly interdisciplinary perspective, particularly one that draws simultaneously on philosophical and technical insights. Some law scholars concerned with digital media have devoted attention to the emerging and contested legal doctrine around protecting individual privacy in the cloud (Anthes 2010; Duffany 2012; Fischer 2012); separately, proposed technical solutions for secure cloud storage are legion (Bowers et al. 2009; Kim et al. 2012; Lopez-Alt et al. 2012; Zhang et al. 2013). However, we see a value in combining insights from the humanities with work from computer science. Bryan Pfaffenberger has termed Science and Technology Studies (STS) "the political philosophy of our time" precisely because, as a discipline, it prompts analysis combining insights from both the human and applied sciences (Pfaffenberger 1992, p. 309). Moreover, Pfaffenberger (1992), Bruno Latour (1992), and others (Nissenbaum 2011a) have argued that the public perception of new technologies like cloud storage as socially and politically neutral and "natural" developments in the computational order of things need disruption in order to prompt critiques and reappraisals. The best way to accomplish this unsettling of perceptions is conceptually and at an early stage in a technology's widespread use by drawing on analyses from multiple disciplines and perspectives.

Taking these injunctions to heart and building on previous scholarship in STS and the philosophy of technology, we suggest that cloud storage as a technological innovation is by no means value-neutral. In order to investigate what values might be inherent and emergent in cloud storage, we conducted our analysis through a design case study exploring one particular issue in cloud computing, that looming question noted above: can users take advantage of the cloud to store data without compromising the privacy of that data? Using the problem of thinking through the meaning of privacy within distributed information systems as a springboard, we proceeded to attack this question both conceptually and technically. To do so, we employed two similar methods for designing with values in mind: Value-Sensitive Design (VSD) (Friedman and Nissenbaum 1996; Friedman et al. 2006), "a theoretically grounded approach to the design of technology that accounts for human values in a principled and comprehensive manner throughout the design process" (Friedman et al. 2006); and Values @ Play, a method similarly invested in "systematically incorporating values in the design process" (Flanagan et al. 2008). As Flanagan, Howe, and Nissenbaum observe, "the study of human and social dimensions of technology is so demanding [because] the areas of knowledge and the methodologies it straddles are traditionally both far-flung and self-contained" (p. 324). Given the challenges inherent in drawing together even as proximate a pair of disciplines as the philosophy of technology and computer science, VSD and Values @ Play seemed the most practical systems at hand. Over the course of the project, we also came to embrace many of the principles of a third approach to designing with values in mind, namely reflective design (Sengers et al. 2005). These three methods all suggested ways to theorize while designing, and design while theorizing.

The end result of our analysis via design was Lockbox, a secure, built-from-scratch, cloud storage system designed through the application of our mixed-methods framework. While stemming from the initial impetus to explore the values space of cloud storage, the design concept for Lockbox evolved to explicitly preserve certain values expressed in conventional cloud storage systems—namely, usability and data availability—while radically improving technical expression of the values of privacy (defined here as the contextually appropriate flow of information (Nissenbaum 2010, 2011b)), autonomy, and user empowerment within the distributed ecosystem of the "cloud." As such, we aim for Lockbox to serve as a model for other privacy/security-aware alternative to popular systems like Dropbox. Johnson (2006) has observed that, "computer systems cannot by themselves be moral agents, but they can be components of moral agency." As a design case study, the Lockbox project seeks to undermine a particularly prevalent false dichotomy (that of privacy vs. usability online), to empower users with an alternative values-aware system, and to prompt reflection on the values inhering in our digitized lives.

Methods

Following the model of previous studies which deploy frameworks such as VSD or Values @ Play to discover, consider, and incorporate values into technology design, our case study and analysis deploys a methodological heuristic made up of certain "constitutive activities" (Flanagan et al. 2008, p. 333), through which abstract values are linked to material artifacts—in this case, the design affordances of online distributed systems. The activities are variously described as using "discovery, translation and verification" (Flanagan et al. 2008) or "value conceptualization, technical analysis and empirical investigation" (Friedman et al. 2006). As noted, in this paper we take a blended approach incorporating both Values @ Play and VSD, since the actions suggested by the first heuristic can be applied to any of the three modes of inquiry proposed by the second. In addition, we take seriously the constructive suggestions of Phoebe Sengers and her coauthors, proponents of what they term "reflective design" (Sengers et al. 2005).

Reflective design, as Sengers and her co-authors describe it, is based on the principle that "technology design practices should support both designers and users in ongoing critical reflection about technology and its relationship to human life" (Sengers et al. 2005, p. 50). The authors admit their indebtedness to other models for designing with values in mind, but propose "critical reflection in and of itself as a core value for technology design" (p. 51). Sengers and her co-authors lay out a series of principles and design strategies for reflective design. These principles include the idea that "designers should use reflection to uncover and alter the limitations of design practice," and further, that designers ought to reflect in order to "re-understand their own role in the technology design process." Moreover, the authors contend that "reflection is not a separate activity from action but is folded into it as an integral part of experience," and that this experiential reflection can ideally result in "a dialogic engagement between designers and users through technology" and its real-world application (pp. 55–56).

We agree that critical reflection is a necessary core value for the design of any new technology, whether intended for research purposes or commercial production (indeed, VSD and Values @ Play could not exist as methods without this type of reflection). Scholars such as Kerr (2010) have warned that habituation to the affordances of digital technologies without such reflection diminishes user capacity to be "the authors of our moral world"; this warning entails a commitment by both designers and users to "maintain the ability to access it [e.g. moral and ethical reflexivity) and make use of it" (p. 303). Moreover, we appreciate Sengers and her co-authors' citation of numerous other such frameworks for human–computer interaction (HCI) (such as ludic design, critical design and reflection-in-action) in the structure of reflective design. We choose to emphasize reflexivity at the outset as a core value underpinning our approach to VSD and Values @ Play as methods, and specifically read the work of Sengers and her coauthors on reflective design as formative in theorizing and designing around values in cloud computing.

In this spirit of reflective design, the design team took note of two important design principles suggested by the work of Internet pioneer David Clark at the outset of the project. The first is that it is important to "design for tussle—for variation in outcome—so that the outcome can be different in different places"; and second, to "modularize the design along tussle boundaries, so that one tussle does not spill over and distort unrelated issues" (Clark et al. 2005, p. 466).

The design team identified "variation in outcome" as a particularly important valence of user autonomy online. By foregrounding user autonomy as a central value to translate into technical affordances, the design team sought to encourage the kind of indeterminacy which ideally supports the positive ethical habituation suggested by Kerr (2010). Though differences in outcome can sometimes be negative, it is rare that entirely undifferentiated outcomes are ever wholly positive: distributed systems must provide a robust technical framework encouraging individual choice in how these systems are actively used.

As Clark and his co-authors note, a second important element of program design that inevitably shapes the translation of values into design practice is modularity. Different pieces of the system (code modules) are generally built separately, and then "networked" or yoked together to create a working application. This design process is typical of an outcome-based approach in programming that seeks particular algorithmic solutions to resolve particular problems using the skills and coding toolkit the particular programmer possesses. As such, consciously considering the values motivating the project, and attempting to engineer a coherence of values across different modules of code, was on reflection a critical task of the design team from the outset, following Sengers et al.'s position that "reflection is not a separate activity from action but is folded into it as an integral part of experience (Sengers et al. 2005, p. 56).

Identifying values

User autonomy

In our preliminary discussions, we as a design team identified the concept of individual online autonomy—what David Clark and his co-authors term "user empowerment" in their seminal 2005 paper (Clark et al. 2005)—as a central value of concern: the desire to explore the prospects for user autonomy and enhance it if possible underpinned the team's general critique of cloud computing's challenge to privacy, our sense of Dropbox's shortcomings in particular, and our intuitions for improvements. According to Clark et al., user empowerment is "the preference that the user, rather than the service provider or the software provider, be able to pick what applications to run, what servers and services to use, and so on." Clark and his co-authors suggest that user empowerment is a basic principle of the Internet itself: it is "the manifestation of the right to choose—to drive competition, and thus drive change" (Clark et al. 2005). In the context of distributed systems such as cloud computing, we take "user empowerment" to refer to the accentuation of accessibility and user autonomy promoted by the cloud's distributed aspects while diminishing the prospects for data to go astray outside a user's individual context, permissions, or choice.

The concepts of user empowerment or autonomy in cloud computing are shaped by the cloud's distributed model of data availability. The user model for cloud storage presumes that the physical mobility of people (across cities, borders, access terminals, or mobile devices) will not necessarily need to match the flow of those individuals' data. Instead, cloud storage services have found wide success based on their assessment that individuals desire a virtual place to store data without requiring the end-user to update or duplicate files constantly, or worry about leaving a particular file on a particular piece of hardware. In other words, users desire a virtual place accessible from anywhere with network access, and feel empowered and autonomous through their own contrasting potential mobility.

Privacy

There is a voluminous literature available on the subject of online privacy in general (Cohen 2000; Kerr and McGill 2007; Lyon 2007; Ohm 2005; Solove 2006), here defined as the "appropriate flow of information" (Cohen 2012; Nissenbaum 2011b), and on privacy in the cloud in particular (Gellman 2009; Jansen and Grance 2011; Pearson et al. 2009; Ryan 2011). As Brian Whitworth and Aldo de Moor (Whitworth and de Moor, 2003) suggest, "Internet privacy concerns seem essentially a conflict between a social requirement (privacy) and current Internet system design"; moreover, the social requirements of online privacy are in flux themselves thanks to technological advances, though there is broad agreement that privacy norms have changed, not disappeared (boyd 2010; Boyles et al. 2012; Nissenbaum 2010). Whitworth and de Moor observe that the perceived conflicts between user expectations around user autonomy, privacy, and usability within digital media technologies in general represent a "social-technical gap [to] the degree software fails to meet social requirements"; this insight suggest the onus is on designers to rise to the challenge of creating more nuanced systems, not on the user to accept wholesale abandonment of particular values such as privacy.

In the specific context of cloud storage, Pearson (2009; Pearson et al. 2009) observes that, "the privacy challenge for software engineers is to design cloud services in such a way as to decrease privacy risk, and to ensure legal compliance." While it should perhaps go without saying, other studies have concluded that threats to a user's privacy "vary significantly with the ToS and privacy policy established by the cloud provider" (Gellman 2009, p. 6) In the case of Lockbox, the challenge with designing privacy affordances lacking in other cloud-based storage services hinged on balancing the usability of the system for end user with a robust encryption regime enabling privacy. Achieving this balance involved rethinking the role of the cloud storage providers like Dropbox as "middlemen" for the content of stored files and with access to user data.

Usability

In the context of digital applications, usability has been defined both as "a quality attribute that assesses how easy user interfaces are to use [and] to methods for improving ease-of-use during the design process" (Lodhi 2010; Nielsen 1993, p. 26). This type of convenience, or rather its lack, is often presented as an unfortunate but inevitable design outcome of online applications relying heavily on encryption. The design team was mindful of this difficulty, and identified the apparent value conflict between privacy and usability as a major challenge to Lockbox as both a reflectively designed test system and as a potentially commercially viable application.

Cost

One final value the design team considered when identifying values at the outset of the Lockbox project was that of low cost, both in terms of financial resources and in terms of computer resources. Given that the political economy of distributed systems ties the latter to the former for users, the team's design goal was to limit both. Much like commercial services like Dropbox, the design team decided to use Amazon S3 as the storage site for Lockbox; while this decision does expose Lockbox to some of the same vulnerabilities of Amazon's TOS, using a service like Amazon S3 costs fractions of pennies to store and retrieve gigabytes of data. As noted above, services such as S3 do lack certain built-in privacy protections; however, building these protections into Lockbox itself would theoretically make this problem moot. The cost of a system built with both privacy affordances and usability in mind was therefore a major question for the Lockbox design: would it be possible to foreground low cost along with the other values identified in this initial phase of the project?

Translating values

User autonomy

Returning to the previously noted definition of user empowerment as "the preference that the user, rather than the service provider or the software provider, be able to pick what applications to run, what servers and services to use, and so on" (Clark et al. 2005), the design team considered what this species of user empowerment would look like in the realm of cloud storage. The business models of many cloud storage services seem ill-suited to providing user empowerment as described above: data is stored on servers without the user's knowledge or control, is viewable to the employees of a service, and is vulnerable to security breaches. It became clear to the design team that the broader challenge of fostering individual user autonomy and empowerment online led to the apparent value conflict between privacy and usability in cloud storage systems.

So considering how to design user empowerment into distributed systems forced the design team to consider the question of autonomy in digital communication as a value in itself. According to one seminal definition provided by Friedman and Nissenbaum (1997), autonomous individuals are "individuals who are self determining: they construct their own goals and values, and are able to decide, plan, and act in ways they believe will help to achieve their goals and promote their values" (p. 466). According to Friedman and Nissenbaum, "user autonomy seems to have less to do with simply the degree of control and more to do with what aspects of an agent are controllable, and the user's conception and knowledge of the agent" (Friedman and Nissenbaum 1997, p. 467). The design team thus made a decision to explore the question of individual autonomy and its relationship to the autonomy of data at its analytic root, as elucidated in the Platonic dialogue *Phaedrus*. The reasons for this decision were two-fold: first, the fact that the problem of data acting autonomously outside an individual's influence was literally an ancient one suggested that finding workable solutions would be challenging, and that a return to first principles would be useful; second, the design team took seriously the first precept of Sengers' et al.'s vision of reflective design, that "designers should use reflection to uncover and alter the limitations of design practice" (Sengers et al. 2005, p. 55). In this case, probing contemporary design problems with ancient philosophical insights seemed a relevant means of discovering some of the team's own limitations in thinking and designing.

In the *Phaedrus*, Plato argues that the autonomy possessed by the individual person is absent from the data a person produces. Plato understands the written word as without "serious intent"; this data can neither "speak [in its] own defense [nor] present the truth adequately" (Plato and Hamilton 1961)(276c). Plato warns that, "once a thing is put in writing, the composition, whatever it might be, drifts all over the place, getting into the hands of not only those who understand it, but also those who have no business with it" (275e). Moreover, the promiscuity of data, whether inscribed on a wax table or encoded in a cloud server, means data is always potentially open to being "ill-treated and unfairly abused" (275e). It is precisely data's fundamental materiality—and hence, mobility—which Plato suggests leaves it open to dissemination, via either an individual text's material transit through time and space, or its mechanical (and in our era, digital) reproduction.

A focus on the material circulation of media and the materiality of a medium's effects on the dynamics of information systems of all sorts is not a novel subject in communications and media studies. German media theorist Friedrich Kittler (1999) has noted that the "general digitalization of information and channels erases the difference between individual media." However, Plato's insight regarding data's reproducibility has particular salience when data is so radically out of the hands of the individual via the cloud.

Instead of data detached from an individual, Plato champions the oral tradition of "living speech" as the ideal mode of discourse. Plato's definition of "living speech" parallels contemporary debates regarding the virtue of "liveness" on television or "real-time" mediation via the World Wide Web. Clearly telephone conversations or videoconferencing do not directly substitute for face-to-face contact. Yet these mediating technologies have universal appeal precisely because they capture some portion of the uniqueness of a particular situational moment. In a contemporary world of viral videos, leaked diplomatic cables and dubious chain emails, data has never been more promiscuous in its circulation and accessibility: the value of "liveness" stems from its continuing ineffability, from the impossibility of perfectly preserving the subjective experience of the passing moments in a live interaction—and the dynamism, for good or ill, which this impossibility demonstrates. Applying Plato's valuation of communicative "liveness" to distributed systems, the individual user of a cloud storage service could be said to be in possession of two distinct kinds of "liveness": their material life offline, and the "liveliness" of their online data. User autonomy and empowerment spring from the coordination and synchronization of both these species of liveness. The “liveness” or perhaps more aptly “liveliness” of data is a key component of structuring user empowerment in cloud storage.

In the context of cloud storage, we took the insight from Plato that we could understand user autonomy and empowerment in the cloud as the correlation between (1) the lived decisions and choices of a user moment-to-moment and (2) the synchronic availability of data within the cloud that was solely legible and available to that particular user when or where she required it. The recognition that user empowerment and autonomy might be produced through the secured, consistently contextually appropriate availability of an individual user’s data is the key “value proposition” of Lockbox. Data in the cloud should always be conveniently accessible from multiple points, matching a user’s potential physical mobility; however, when a user is not accessing data, a user’s autonomy is only truly preserved, and her empowerment sustained, if that data is functionally inert and inaccessible to others.

As Canadian legal scholar Ian Kerr observes in his (2010) analysis of digital rights management (DRM) systems, designing for user autonomy and empowerment also has an ethical dimension. Kerr points out that the technical affordances of digital systems participate in what Kerr describes as “an attempt to promulgate the ‘automation of virtue’” (p. 288): the capacity to choose certain ethical or moral decisions are removed from the ken of individual user choice and designed into information systems themselves. Kerr suggests that any effort to automate virtue in information systems is doomed to failure, noting Aristotle’s insight that “the moral attainment of virtue relies fundamentally on practicing, or developing, the virtues in real situations over the course of a lifetime” (p. 290).

In terms of cloud storage, the current paradigm entails precisely the opposite of what Plato and others saw as autonomous or empowered communication. Aristotle’s view of habituation is consonant with Plato’s valorization of “live,” face-to-face discourse: in both cases, ethical behavior stems from a constant internal debate within individuals that is both constitutive of moral autonomy, and the key of moral autonomy’s perpetuation through action. As Kerr observes in the context of DRM, if the automation of virtue in information systems disinclines a user from “negotiating with herself about what honesty entails or from deciding what she will morally permit herself to do, her ability and desire to cultivate practical wisdom and the achievement of moral excellence will be impaired.” “Such users,” Kerr concludes, “will become morally disabled” (Kerr 2010, p. 294) precisely because the scope of the individual user’s autonomy will be limited. The same problem applies in the case of the cloud: user autonomy and empowerment have the potential to fail when user data is so detached from the user herself.

Privacy versus usability

One way through which the problem of user autonomy or empowerment is often expressed in design is the purported tradeoff between information privacy and convenience/usability. Building on Nissenbaum’s (2010, 2011b) work on privacy in an online context, we take it as an important axiom that users gain more autonomy, and hence empowerment, when they have the maximum flexibility possible in determining the appropriateness of the flow of their own personal information. This flexibility can be designed into the user’s experience by ensuring that data is by default only accessible according to the wishes of the user in any particular moment.

Privacy is one way in which user autonomy is expressed in everyday life. In the case of cloud storage, designing with the value of privacy in mind would involve design choices that facilitate the “appropriateness” of the information accessible to and about the user (Itani et al. 2009). In theory, data is mobilized only when the user accesses it; however, the security problems and privacy concerns noted in our Introduction suggest data stored in the cloud is still too mobile, too “promiscuous,” and too often subject to inappropriate use or abuse. When privacy is understood as contextually appropriate flows of this information, however, it becomes clear that user autonomy online is a function of how nimble a system can be at enabling contextually-appropriate data access, while also being usable and convenient from the perspective of the end user.

In order to facilitate user autonomy and empowerment through enhanced privacy protections, it is necessary to restrict the “promiscuity” of encoded data—to more effectively immobilize it, working to ensure that the data is “lively” only when a user is “live” on the network. Synchronizing the “liveness” of data online with the input of the live user ties decisions about stored online data to an individual. The “safety” of data stored correlates to how inert—or dormant—that data might be when it is not being accessed by the user: how safe it is from tampering or misuse by outside actors, whether malicious or not. Indeed, this model of “liveness” neatly aligns Plato’s view with Clark et al.’s (2005) valorization of user empowerment on the Internet: maximum individual autonomy and minimum data autonomy ensure the highest possible degree of empowerment for the individual user.

In defining user autonomy as a blend of privacy questions and usability concerns, the design team turned to cryptography as a possible technical solution to enable both strong privacy protections and fair usability standards.

Applied cryptography in the context of the cloud is hardly a new idea (Chen et al. 2012; Wang et al. 2011). However, the creative use of encryption as a way to foster appropriate information flows has been bolstered by the insight that privacy, and hence autonomy, can be enhanced through selective techniques of obfuscation (see Brunton and Nissenbaum 2011), particularly in cases where users have little say otherwise in how their data flows are channeled and appropriated.

Any digitally encoded data is in some sense mobile—it can be moved or copied easily, including between different servers or terminals networked to the cloud. However, the contextual salience of encrypted data—its legibility, and hence its potential for use and abuse—is highly curtailed. Flows of information persist, but their contents are blocked from those without a decryption key. As a result, the user of a system like Lockbox is able to view her data as readable text, while unauthorized actors peering at data within the cloud storage server or network would perceive a user's data as ciphered text. The ongoing design challenge with Lockbox is to make it relatively easy to use. Usability is not always at the center of discussions around privacy protection, but should be: if a test of usability is that a product "enables the intended users to achieve the intended tasks" (Bevan 1995), then by definition the onus is on designers to consider not just what task needs to be achieved, but also how that task is carried out—easily, but also in a way that protects a user's privacy.

Cost

As with all designs that seek to incorporate certain values, the proverbial devil is in the technical details, and the challenges inherent in making a design both practicable and usable. In this case, getting the balance between privacy and usability right to enhance user autonomy comes with costs: not only the work involved in designing Lockbox, but also the user's time in navigating the intricacies of key encryption and decryption, about which more will be said below. The challenges of creating a cryptographic application that is both secure and user-friendly are considerable, but not impossible. If nothing else, the development of Lockbox exposed the magnitude of this design problem anew and prompted further reflection on the challenge. While the solutions offered below are partial and preliminary, our analysis suggests that while the cost of developing such a system may be high at the outset, its value as an ongoing resource for individual users is also high. Autonomy, in other words, is worth its weight in gold and hard work.

The dynamics of data mobility and user autonomy we describe above suggest that individual users of distributed systems should be afforded the ability to mobilize their data selectively and contextually on moral as well as practical grounds. More broadly, thinking through "user empowerment" and "liveness" together allowed the design team to attempt a balance of privacy and usability in the actual design of Lockbox; we elaborate on this process of designing for values below.

Designing for values

User autonomy

Throughout the design process, the Lockbox team was cognizant that its technical choices regarding privacy protections and usability would condition the level of autonomy the system's users would enjoy. These design choices began with the programming language itself: it was important to the design team that Lockbox be portable across platforms and operating systems. As such, Lockbox was written using the versatile Python programming language, licensed under an open-source Sleepycat License, and designed so that its configuration by users could be accomplished through a web browser.

Lockbox was designed to operate as a background user program, automatically synchronizing the data stored in file directories specified by the user to a cloud storage service. This data is encrypted both in transit to, and in storage on, the cloud service provider's servers. The design of Lockbox also enables future versions of the program to let users share encrypted data with friends who also use the service, and makes this secure file sharing private and always available.

In conceptualizing Lockbox, the design team was inspired to consider the role of encryption in enhancing online autonomy in several novel ways. Lockbox's encrypted storage is prototypical of a relatively new approach to cloud storage (Wang et al. 2011). This approach is one that sets the parameters for online data to be protected within a particular encrypted "slice" of the "cloud" controlled by a particular user. As a test system, Lockbox was designed to begin to carve out autonomous spaces protected by encryption, yet still legible to particular individuals.

As with any online application, absolute user autonomy cannot be assured simply by using a system like Lockbox. Lockbox assures users that through its algorithm they are secure and free from harm in the privacy of their data storage. While security is not zero sum, the data that is secured sometimes is: the secure and encrypted nature of Lockbox's service may prompt care and attention from lawmakers and regulators with an interest in tracking and classifying in the "cloud." Moreover, Lockbox obviously does not protect side-channel or covert channel attacks, such as screen shots or password copying — threats which are predicated on a breakdown of trust offline.

Privacy and usability

Along with its mobilization for industrial and military applications, cryptography has a long history as a technical privacy solution (Levy 1996). However, the privacy policies of services like Dropbox raise concerns around middleman companies that merely use symmetric key encryption. Symmetric key cryptography is very similar to the everyday use of passwords to log into computers or email services: to access a password-encrypted document simply requires the correct password. However, this password, or "key," must be shared between all parties who the owner wants to grant access to the data; in existing cloud storage systems, one of the most common and practical methods for securely storing data comes through using a symmetric key encryption method known as the AES-256 block cipher.

Because a company like Dropbox itself stores the password that is also used to encrypt the data stored by the same organization, Dropbox can access the content of the data stored via their services at any time. These companies can also release this data in readable text when pressured by an outside agency, at their employees' whim, or when their security services are compromised. Current best practice for users who want to combine the convenience of Dropbox with some type of stand-alone encryption system involves an application called TrueCrypt, a free and open source disk encryption system. However, this combination of systems is not always easy to use (for instance, the order in which the TrueCrypt and Dropbox are started (or stopped) when a computer boots (or shuts down) can effect the integrity of the data and of the encryption itself).

Instead, the Lockbox design team turned to a cryptographic design based on both symmetric key and public key cryptography. Public key cryptographic techniques ask the user to both "sign" and encrypt data and are considered stronger than symmetric key systems for two reasons: first, signing data (with a private key) guarantees that the recipient knows that sender of the data is who the sender claims to be; second, encrypting data (with a public key) guarantees that only the corresponding private key owner can decrypt the data. The public encryption key cannot be used to decrypt the data encrypted by the public key.

The design team decided that the best possible balance of privacy protection and practicable usability came through implementing this hybrid cryptosystem, a conceptually well-know but practically under-utilized combination of symmetric key and public key cryptography, to ensure privacy for user's files. Hybrid cryptography works as follows: a randomly generated 32-character password is generated for every file to be encrypted; the password is then used to encrypt the file in the aforementioned AES-256 format. This password is then itself encrypted with a public key (either that of the user or, in the case of Lockbox's prospective sharing function, of whomever the file owner wants to share the file with). The public key-encrypted password files (more than one, since there are multiple users to share the file with) and the symmetric key-encrypted file are then all uploaded to the cloud service.

The file encrypted with the symmetric key and the corresponding password file or files encrypted with the public key are stored and made accessible to the user when she accesses the service. In future iterations of Lockbox, these files could also be shared to whomever the file owner wants to share the original data with.

Perhaps predictably, managing the exchange of users' public keys remains the central challenge to providing contextually appropriate privacy while also allowing the system to be usable along the lines of other current applications like Dropbox. Some of the issues within this broader problem include:

- the design for how to appropriately handle access revocation (whether to retroactively apply the revocation to all previous versions of a file, or not)
- how and where to store public keys and access control lists (ACLs) (in the cloud or on a more secure server).
- an associated risk of revealing information about users (therefore compromising privacy) with a bad design of key and ACL management.

At the time of writing, the initial Lockbox prototype uses a naive, proof-of-concept privacy, key management, and access control system: for every file and file update, a new password key must be generated, and revocation is accomplished changing permissions to the corresponding object, as well as removing the revoked user's keys from the list of keys with which to encrypt the files. The choice to implement a simple yet inefficient design was made by the design team for the purpose of demonstrating a workable proof-of-concept prototype. Nevertheless, the design team has debated the "correct" design of the key management and access control system with the balance between privacy and usability in mind; we expand on this point below in the "Cost" section.

Nonetheless, deploying a hybrid cryptosystem within the cloud allows Lockbox to build the view of privacy based on contextual access and flow of data to a particular user into the design of Lockbox. The design team took inspiration from other privacy-enhancing technologies (PETs); Herbert Burkert (1997) has defined PETs as technologies that "seek to eliminate the use of personal data altogether or… give direct control over revelation of personal information to the person concerned" (p. 127). Burkert's taxonomy of PETs includes subject-oriented, object-oriented, transaction-oriented and system-oriented technical concepts; Lockbox operates primarily as a subject-oriented PET, by aiming to "eliminate or substantially reduce the capability to personally identify the acting subject" through hybrid cryptography (p. 125). The design team is cognizant of Burkert's critiques of PETs, particularly that the values underpinning the design of such technologies must be carefully though through: our analysis around user autonomy as a function of both privacy and usability provides one metric among many for considering the efficacy of PETs in general, and in the cloud in particular (Hon et al. 2011).

Cost

In an effort to minimize financial costs both to the design team and to end users running Lockbox, the design team has continually searched for cost-effective design solutions. Encryption has costs both in terms of the system resources used and the amount of time it takes for a system to process data. However, technical innovations taking advantage of new research can sometimes help on both counts. An example of one such design decision in Lockbox leverages the rsync algorithm, in addition to Amazon Web Services' simple queue service (SQS) and SimpleDB (Tridgell and Mackerras, 1996). "Rsync" is a well-known algorithm designed to quickly compute differences between files and quickly apply updates to files. SQS is designed to store messages as they travel between computers. SimpleDB is designed as a (optionally, strongly consistent) database optimized for efficient index and select queries (non-relational queries). By using rsync in its design the design team has attempted to maintain the original vision of a correctly-implemented file sharing, syncing, and versioning while minimizing both financial and usability costs, though more work remains to be done.

Outcomes

Lockbox has yet to undergo significant testing by users, and more feedback is needed to assess the degree to which the design team has been successful in designing a cloud storage service that brings the values of user autonomy and empowerment to the fore through balancing privacy with a usable experience. We are mindful of Sengers et al.'s six strategies for putting reflective design into practice, particularly the need to use technical design to "probe" for unassumed user needs and the exploration of previously overlooked concepts and metaphors brought forward by the user" (Sengers et al. 2005, pp. 56–57). As such, the design team has identified a number of areas where empirical data will be particularly critical to evaluating its design choices; these include the following:

- Managing public key distribution is a hard—perhaps the hardest—technical problem for Lockbox to solve, and the issue that most explicitly highlights the difficulty of balancing privacy and usability within the design of this or any encrypted system. For instance, the design must be able to appropriately handle access revocation and, more importantly, personal key recovery; other concerns include determining how to store public keys and ACLs on the cloud, if at all. At the moment, the system for trading public keys is involved and relatively inconvenient, relying on encrypted email messages that themselves pose security challenges. Balancing the risk of revealing information about users (and therefore compromising privacy) with a more usable design of key and ACL management is perhaps the central user experience challenge facing the design team.
- In a related vein, Lockbox's system for trading public keys is based on a trust model that assumes, in particular, that users will trust their collaborators not to share or otherwise expose their public keys. Given these constraints, sharing files through Lockbox would ideally be suited for small groups of users with relatively robust bonds of trust. However, given the scalability of the storage system itself, even large institutions that need to back up data and provide easy semantics for data privacy and sharing amongst an institutions' members could benefit from such a service, provided the number of actual users was relatively small. Whether Lockbox would prove useful for large groups is another question the design team seeks to answer through alpha testing.
- In thinking conceptually through questions of data legibility and user autonomy, it is clear that encrypted data, while losing much of its symbolic "sense" to the eyes of a human user, is still quantifiable and traceable (Kerr and McGill 2007). This data can be subject to analysis of its quantity and patterns of circulation, among other factors. Lockbox does not make an attempt to obfuscate user metadata, nor the patterns of data access left by users; while observation of these variables may lead data to be exploitable by some malicious actors, we consider Lockbox's encryption capacities a potent defense against the majority of data breach threats. And of course, Lockbox is not able to protect against security breaches in cases of hardware loss—for instance, losing a laptop might permit malicious actors access to "audit" file access patterns. We understand the values "trade-off" involved to be

between availability and auditing (itself a privacy concern). Feedback regarding the priority of these security issues from users will provide valuable empirical data in order to determine what further security features should be incorporated into future versions of the product.

- Lockbox does not currently enable any mechanisms to track, or audit, a file after it has been decrypted on a person's computer (see Geambasu et al. 2011). The current lack of auditing in Lockbox also questions the possible promiscuity of a user's private key: the design team acknowledges that should one private key be used by multiple individuals, a malicious person could then expose the paired public key encrypted files to several "bad actors." In the estimation of the design team, the current Lockbox threat model nonetheless empowers end-users' privacy over competing products.

More broadly, interactions with Lockbox's prospective users continue to influence possible future design considerations. While the design team is aware that new technologies will inevitably breed new uses and new users, specific user constituencies have already been identified. One particular user group already expressing interest in a service like Lockbox is the human rights community: organizations needing a secure way to transfer and share data regarding human rights abuses in authoritarian regimes have noted Lockbox's utility. Discussions with technologists who work with human rights workers have yielded valuable insight into needs that Dropbox fails to meet. For instance, human rights activists who must share large videos with individuals outside of a country of interest require bandwidth-shaping features to avoid detection by governments who censor network activity. Without bandwidth shaping to limit the amount of traffic uploaded or downloaded through a file sharing service, giant bursts of traffic could raise red flags for censoring governments. One suggested feature to improve Lockbox's use value to this demographic is the ability to manually limit the rate of file transfer (for slow bandwidth): this feature would be necessary in order to protect large uploaders from "standing out," and begins to address questions about obfuscating the volume of transmitted data. One technical solution under investigation by the design team is discovering a means to ensure that Lockbox operate through a SOCKS proxy, similar to a technique applied by Tor ("Tor Project: Overview," n.d.).

Reflections and conclusion

While Lockbox is in an early stage as a mass-market application, it has already proven fruitful for mapping out some of the challenges and possibilities of translating values into IT design. Consonant with Sengers et al.'s contention that "designers should use reflection to re-understand their own role in the technology design process" (Sengers et al. 2005, p. 55), our own focus on looking back at, and reflecting on, Lockbox's design has helped us identify several key areas for further investigation.

Reflective design, design reflection

As noted, the design team attempted to follow two key important design principles taken from Clark et al. (to "design for tussle [and] for variation in outcome," and to "modularize design" (Clark et al. 2005, p. 466)) throughout the design of Lockbox. Our reflection on the design experience produced a number of other related insights.

First, it is important to note that designers authentically desire their own values to align with the values of their users. This goodwill can be treacherous, however, if assumptions about the values, capacities and empowerment of end users are made uncritically or unreflectively. The values of designers and users may not entirely overlap even in cases in which designers do reflect deeply on the underlying values of the systems they create. Through reflection and discussions at the conclusion of the project, the design team identified a number of discontinuities and assumptions separating the process of identifying and translating values into design elements from the technical minutiae of building a workable experience for users.

In many ways, the difficulties of finding value commensurability between designers and users mirrors the challenge of providing the users of distributed systems autonomy—and ultimately empowerment—within the economic, technical and institutional constraints of information technology design. Indeed, the question of value commensurability is often shifted "to the other foot": advocates for enhanced privacy online, for example, are sometimes accused by their detractors of forcing a particular values agenda on consumers who, through their actions, are claimed to have given up much interest in online privacy. We believe that designers should be particularly aware and reflective of the full spectrum of their potential users, and advance the technical means to provide the kind of choice Clark et al. (2005) advocate for: once again, we advance user empowerment as a basic principle for design online precisely because choice, while not a panacea, is a necessarily if not sufficient condition for the very possibility of ethical action.

Another consideration shaping designer values—one unique to the academic context—involves the research and design priorities of university computer science departments and scholars. Whereas previous studies suggest that incorporating VSD into design communities in the business world is challenging given the difficulties of squaring particular values frameworks with the exigencies of contemporary capitalism (cf. Manders-Huits and Zimmer 2009), academic designers face a different challenge. Given the research-oriented focus of the university, academic computer science is constantly in search of novel research techniques, designs and theories: research that is primarily synthetic—though combining previous insights in novel ways—is not always valued. This dynamic also depends on the particular subfield of computer science in which designers are working—for instance, the academic pressures and priorities in system design and analysis may be quite different from those involved in HCI. In contrast, while there is considerable commercialization of computer science research, these products do not necessarily accomplish novel ends, nor do they necessarily consider values and design carefully. The design team was motivated to simultaneously work on theoretical problems considered "state of the art" in network research while still attempting to create usable software—while, in the case of Lockbox, *also* attempting to incorporate into Lockbox a high degree of reflexivity and responsiveness to values discovered through the design process.

Conclusion

As a potential mass-market system, Lockbox is certainly not alone. A number of consumer applications currently provide user storage in the "cloud": these include SugarSync, Mozy, Apple's iCloud service, and market leaders Dropbox and Google Docs. These applications do claim to encrypt user data (though as previously noted, they tend to do so only external to the applications themselves: the data is still vulnerable to inadvertent or deliberate access by employees of the companies involved, or to malicious actors hacking into the system). Moreover, a number of stand-alone consumer applications featuring enhanced security are also available, some of which are similar conceptually to Lockbox: these include Wuala, Tahoe-LAFS, Lockify, and SpiderOak. Arguably, these systems sacrifice the convenience of a single folder abstraction (e.g., Wuala) or demand extraordinary expertise and confidence from end-users (e.g., installing a new file system). However, secure cloud storage is a crowded commercial space; recent controversies such as the growing furor around the National Security Agency (NSA) will no doubt exacerbate competition and innovation in the field.

The Lockbox design team determined that a new cloud storage application build specifically with values in mind was a salutary project not simply to provide yet more competition in an already crowded design space. More central to the project was a desire to work through the value questions implicit in distributed "cloud" systems, and proceed through values-aware design heuristics towards a workable end product. Further empirical analysis to systematically identify the design choices made by the other secure cloud storage services mentioned above would also be helpful to future versions of Lockbox: whether these applications have made similar or divergent design choices from those of the Lockbox team, and how design choices have affected expressed values in the use of the application, is a subject for ongoing research.

Finally, the design team working on Lockbox has consistently been aware of the imperative to consider how the values designed into digital systems interact with broader social and more narrowly legal norms. Some of the questions posed by this awareness include asking who is liable for encrypted stored data that may be illicit or illegal; what copyright or export issues, if any, need to be taken into account; and, more generally, what are the underlying ethical tradeoffs involved in securing online privacy for a wide range of individuals, some of whom might use their privacy for morally questionable or even repugnant ends?

An initial response to this last question might draw on the argument regarding the dynamics of data mobility, user autonomy, and the costs of privacy described earlier in this paper. Human life cannot usefully be separated into online and offline components; as Nissenbaum (2011b) has argued, designers of laws and devices should begin their legal and technical work with a commitment to investigate core democratic values such as freedom or privacy, and construct online norms consistent with these values, reinforcing "moral imperatives" stemming from values such as user autonomy, as opposed to technological imperatives stemming or subsumed values of a less democratic bent. At this point, further user research and testing is a necessary compliment to the theoretical and technical work already accomplished to further understand how these ethical questions play out in the real-world context of cloud storage. Ideally, however, the Lockbox project will not only stimulate further conceptual and theoretical work on the status of the individual and her privacy in the "cloud," but also inspire others to design with attention to the values at play within it.

References

Anthes, G. (2010). Security in the cloud. *Communications of the ACM, 53*(11), 16–18. doi:10.1145/1839676.1839683.

Bauman, Z. (2007). *Liquid Times: Living in an Age of Uncertainty*. Malden, MA: Polity Press.

Bevan, N. (1995). Usability is Quality of Use. In *Presented at the Proceedings of the 6th International Conference on Human Computer Interaction (CHI)*, Yokohama, Japan.

Bowers, K. D., Juels, A., & Oprea, A. (2009). HAIL: a high-availability and integrity layer for cloud storage (pp. 187–198). In *Presented at the Proceedings of the 16th ACM conference on*

Computer and communications security, New York, NY, USA: ACM. doi:10.1145/1653662.1653686.

Boyd, D. (2010). *Making Sense of Privacy and Publicity*. Austin, TX: SXSW.

Boyles, J. L., Smith, A., & Madden, M. (2012). *Privacy and Data Management on Mobile Devices*. *Pew Research Center's Internet & American Life Project* (pp. 1–19). Washington, D.C.

Brunton, F., & Nissenbaum, H. (2011). Vernacular resistance to data collection and analysis: A political theory of obfuscation. *First Monday, 16*(5), 1–22.

Burkert, H. (1997). Privacy-Enhancing Technologies: Typology, critique, vision. In P. E. Agre & M. Rotenberg (Eds.), *Technology and privacy: The new landscape* (pp. 125–142). Cambridge, MA: The MIT Press.

Chen, J., Wu, X., Zhang, S., Zhang, W., & Niu, Y. (2012). A Decentralized Approach for Implementing Identity Management in Cloud Computing (pp. 770–776). In *Presented at the International Conference on Cloud and Green Computing (CGC), IEEE*. doi:10.1109/CGC.2012.118.

Clark, D. D., Wroclawski, J., Sollins, K. R., & Braden, R. (2005). Tussle in cyberspace: Defining tomorrow's internet. *IEEE/ACM Transactions on Networking, 13*(3), 462–475. doi:10.1109/TNET.2005.850224.

Cohen, J. E. (2000). Examined lives: Informational privacy and the subject as object. *Stanford Law Review, 52*(5), 1373–1438.

Cohen, J. E. (2012). *Configuring the Networked Self: Law, Code, and the Play of Everyday Practice*. New Haven, CT: Yale University Press.

Constantine, D. (2012). Cloud computing: The next great technological innovation, the death of online privacy, or both. *Georgia State University Law Review, 28*(2), 499–528.

Duffany, J. L. (2012). Cloud Computing Security and Privacy. In *Presented at the 10th Latin American and Caribbean Conference for Engineering and Technology (*pp. 1–9) Panama City, Panama.

Fischer, P. E. (2012). Global standards: Recent developments between the poles of privacy and cloud computing. *JIPITEC, 1*(3), 33–59.

Flanagan, M., Howe, D. C., & Nissenbaum, H. (2008). Embodying Values in Technology: Theory and Practice. In J. van den Hoeven & J. Weckert (Eds.), *Information Technology and Moral Philosophy* (pp. 322–353). Cambridge, UK: Cambridge University Press.

Friedman, B., & Nissenbaum, H. (1996). Bias in Computer Systems. *ACM Transactions on Information Systems, 14*(3), 330–347.

Friedman, B., & Nissenbaum, H. (1997). Software Agents and User Autonomy. In *Presented at the AGENTS '97: Proceedings of the first international conference on Autonomous agents* (pp. 466–469) New York.

Friedman, B., Kahn, P. H., & Borning, A. (2006). Value sensitive design and information systems. In B. Schneiderman, P. Zhang, & D. Galletta (Eds.), *Human-Computer Interaction in Management Information Systems: Foundations* (pp. 348–372). New York: M.E. Sharpe, Inc.

Gain, B. (2011, April 23). Why Dropbox's Privacy Policy Is OK (Just Proceed Carefully). *pcworld.com*. Retrieved April 30, 2011, from.

Garon, J. M. (2011). *Navigating through the Cloud—Legal and Regulatory Management for Software as a Service*. NKU Chase Law & Informatics Institute.

Geambasu, R., John, J. P., Gribble, S. D., Kohno, T., & Levy, H. M. (2011). Keypad: An Auditing File System for Theft-Prone Devices. In *Presented at the EuroSys'11* (pp. 1–15) Salzburg, Austria.

Gellman, R. (2009). *Privacy in the Clouds: Risks to Privacy and Confidentiality from Cloud Computing* (pp. 1–26). World Privacy Forum.

Governor, J. (2011, April 20). My thoughts on Dropbox, corporate and personal privacy and ToS changes. *James Governor's Monkchips*. Retrieved May 6 2013, from http://redmonk.com/jgovernor/2011/04/20/my-thoughts-on-dropbox-corporate-and-person.

Hon, W. K., Millard, C., & Walden, I. (2011). The problem of "personal data" in cloud computing: What information is regulated?—the cloud of unknowing. *International Data Privacy Law, 1*(4), 211–228.

Itani, W., Kayssi, A., & Chehab, A. (2009). Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures. In *Presented at the International Conference on Dependable, Autonomic and Secure Computing (DASC), IEEE*. (pp. 711–716) doi:10.1109/DASC.2009.139.

Jansen, W., & Grance, T. (2011) *Guidelines on Security and Privacy in Public Cloud Computing* (pp. 1–80). Gaithersburg, MD: National Institute of Standards and Technology, U.S. Department of Commerce.

Johnson, D. G. (2006). Computer systems: Moral entities but not moral agents. *Ethics and Information Technology, 8*, 195–204.

Kerr, I. (2010). Digital locks and the automation of virtue. *"Radical extremism" to "Balanced Copyright": Canadian Copyright and the digital agenda* (pp. 247–303). Toronto: Irwin Law.

Kerr, I., & McGill, J. (2007). Emanations, Snoop Dogs and Reasonable Expectations of Privacy. *Criminal Law Quarterly, 52*(3), 392–431.

Kim, B. H., Huang, W., & Lie, D. (2012). Unity: secure and durable personal cloud storage. In *Presented at the Proceedings of the 2012 ACM Workshop on Cloud computing security workshop* (pp. 31–36) New York, NY, USA: ACM. doi:10.1145/2381913.2381920.

Kittler, F. A. (1999). *Gramophone, Film, Typewriter*. (G. Winthrop-Young, Trans.). Palo Alto, CA: Stanford University Press.

Latour, B. (1992). Where are the missing masses? The sociology of a few mundane artifacts. In W. E. Bijker & J. Law (Eds.), *Shaping technology/building society: Studies in sociotechnical change* (pp. 225–258). Cambridge, MA: The MIT Press.

Levy, S. (1996). Crypto Rebels. In P. Ludlow (Ed.), *High noon on the electronic frontier* (pp. 185–205). Cambridge, MA: The MIT Press.

Lodhi, A. (2010). Usability Heuristics as an Assessment Parameter: for performing Usability Testing.*In Presented at the 2nd International Conference on Software Technology and Engineering(ICSTE)* (pp. 256–259) San Juan, Puerto Rico.

Lopez-Alt, A., Tromer, E., & Vaikuntanathan, V. (2012). On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In *Presented at the Proceedings of the 44th symposium on Theory of Computing* (pp. 1219–1234) New York, NY, USA: ACM. doi:10.1145/2213977.2214086.

Lyon, D. (2007). Data, Discrimination, Dignity. In *Surveillance Studies: An Overview* (pp. 179–197). Malden, MA: Polity.

Manders-Huits, N., & Zimmer, M. (2009). Values and pragmatic action: The Challenges of Introducing Ethical Intelligence in Technical Design Communities. *International Review of Information Ethics*, 1–8.

Matthews, L. (2011, April 21). Dropbox responds to privacy outrage. *Geek.com*. Retrieved May 6 2013, from http://www.geek.com/news/dropbox-responds-to-privacy-outrage-1345235/.

Mell, P., & Grance, T. (2011) *The NIST Definition of Cloud Computing (Draft)* (pp. 1–7). National Institute of Standards and Technology, U.S. Department of Commerce.

Nielsen, J. (1993). What is usability. *Usability engineering* (pp. 23–49). San Diego, CA: Academic Press.

Nissenbaum, H. (2010). *Privacy in context: Technology, policy, and the integrity of social life*. Palo Alto, CA: Stanford Law Books.

Nissenbaum, H. (2011a). From Preemption to Circumvention. *Berkeley Technology Law Journal, 26*(3), 1367–1386.

Nissenbaum, H. (2011b). A Contextual Approach to Privacy Online. *Daedalus, 140*(4), 32–48.

Ohm, P. (2005). The Fourth Amendment Right to Delete. *Harvard Law Review, 119*, 10–18.

Pearson, S. (2009). Taking Account of Privacy When Designing Cloud Computing Services. In *Presented at the ICSE'09 Workshop* (pp. 44–52).

Pearson, S., Shen, Y., & Mowbray, M. (2009). A privacy manager for cloud computing. *Cloud Computing* (pp. 90–106). New York: Springer.

Pfaffenberger, B. (1992). Technological Dramas. *Science, Technology and Human Values, 17*(3), 282–312.

Plato, & Hamilton, E. (1961). Phaedrus. In E. Hamilton & H. Cairns (Eds.), *The collected dialogues of Plato* (pp. 475–525). Princeton, NJ: Princeton University Press.

Ryan, M. D. (2011). Cloud computing privacy concerns on our doorstep. *Communications of the ACM, 54*(1), 36–38. doi:10.1145/1866739.1866751.

Sengers, P., Boehner, K., David, S., & Kaye, J. ". (2005). Reflective Design. In *Presented at the Proceedings of the 4th decennial conference on Critical computing: between sense and sensibility* (pp. 49–58) New York, NY, USA: ACM. doi:10.1145/1094562.1094569.

Soghoian, C. (2011, April 12). How Dropbox sacrifices user privacy for cost savings. *slight paranoia*. Retrieved July 10 2013, from http://paranoia.dubfire.net/2011/04/how-dropbox-sacrifices-user-privacy-for.html.

Solove, D. J. (2006). A taxonomy of privacy. *University of Pennsylvania Law Review, 154*(3), 477–564.

Tor Project: Overview. (n.d.). Tor Project: Overview. *torproject.org*. Retrieved May 6 2013, from https://www.torproject.org/about/overview.html.en.

Tridgell, A., & Mackerras, P. (1996). *The rsync algorithm* (No. TR-CS-96-05) (pp. 1–8). The Australian National University.

Wang, C., Wang, Q., & Ren, K. (2011). Towards Secure and Effective Utilization over Encrypted Cloud Data. In *Presented at the 31st International Conference on Distributed Computing Systems Workshops (ICDCS Workshops)* (pp. 282–286) IEEE. doi:10.1109/ICDCSW.2011.16.

Whitworth, B., & de Moor, A. (2003). Legitimate by design: Towards trusted socio-technical systems. *Behavior and Information Technology, 22*(1), 31–51.

Zhang, Q., Luo, B., Shi, W., & Almoharib, A. M. (2013). CloudSafe: Storing Your Digital Asset in the Cloud-based Safe.