Systematizing "Accountability" in Computer Science (Version of Feb. 17, 2012)[1]

Joan Feigenbaum
Aaron D. Jaggard
Rebecca N. Wright
Hongda Xiao

Technical Report YALEU/DCS/TR-1452
February 2012

We provide a systematization of approaches to accountability that have been taken in computer-science research. Toward this end, we categorize these approaches along the axes of time, information, and action; within each of these axes, we identify multiple questions of interest.

Different researchers have (explicitly or implicitly) used "accountability" to mean different things. Our systematization contributes an articulation of the definitions that have been used in computer science (sometimes only implicitly); it also contributes a perspective on how these different approaches are related.

Approved for public release: distribution unlimited
Keyword: Accountability

[1] This material is based in part upon work supported by the Defense Advanced Research Agency (DARPA) and SPAWAR Systems Center Pacific, Contract No. N66001-11-C-4018. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Agency (DARPA) and SPAWAR Systems Center Pacific.

Systematizing "Accountability" in Computer Science

Joan Feigenbaum*, Aaron D. Jaggard†, Rebecca N. Wright‡, and Hongda Xiao§
*Department of Computer Science
Yale University
New Haven, CT USA 06520-8285
Email: joan.feigenbaum@yale.edu
†Department of Computer Science
Colgate University
Hamilton, NY USA 13346
Email: adj@dimacs.rutgers.edu
‡DIMACS and Department of Computer Science
Rutgers University
Piscataway, NJ USA 08854-8019
Email: rebecca.wright@rutgers.edu
§Department of Electrical Engineering
Yale University
New Haven, CT USA 06520-8267
Email: hongda.xiao@yale.edu

***Abstract*—We provide a systematization of approaches to accountability that have been taken in computer-science research. Toward this end, we categorize these approaches along the axes of time, information, and action; within each of these axes, we identify multiple questions of interest.**

Different researchers have (explicitly or implicitly) used "accountability" to mean different things. Our systematization contributes an articulation of the definitions that have been used in computer science (sometimes only implicitly); it also contributes a perspective on how these different approaches are related.

I. Introduction

Traditionally, computer-science researchers have taken a *preventive* approach to security and privacy in online activity: Passwords, authentication protocols, and other before-the-fact authorization mechanisms are designed to prevent users from violating policies and to obviate the need to adjudicate violations and punish violators. Purely preventive approaches to security and privacy have proven to be inadequate as more and more daily activity moves online, and users in different administrative domains must exchange information and transact business without the benefit of a common set of policies and credentials. Many information-security researchers have thus sought *accountability mechanisms* to complement preventive mechanisms. Despite widespread agreement that "accountability" is important in online life, it is not yet a unified research area. Indeed, the word is used by different researchers to mean different things and is not always defined explicitly.

It is our thesis that the lack of agreement about definitions and formal foundations is impeding progress on accountability research and adoption of accountability technology. We offer this systematization as a step toward remedying this situation. Our starting point is a succinct, high-level perspective on the appropriate focus of accountability work in computer science: Accountability mechanisms should enable actions to be tied to consequences and, in particular, enable violations to be tied to punishment. Guided by that fundamental goal, we categorize existing work on accountability along three axes: time, information, and action.

With respect to time, we consider five standard approaches to violations and potential violations of security policies: prevention, detection, evidence, judgment, and punishment. Roughly speaking, these approaches can be linearly ordered in time. First, one tries to prevent violations. When that cannot be done, the goal is to detect violations. If a violation is detected, or even suspected, it may be necessary to gather evidence that can later be used to render a judgment about precisely what happened and whom or what to blame. Finally, actions can be tied to consequences by meting out punishment to the violator. A single accountability mechanism can address one or more of these five phases; most do not address them all. A stand-alone authentication or authorization mechanism that is purely preventive should not be called an "accountability" mechanism, but before-the-fact authorization can be part of a larger system that also addresses the later phases of accountability.

Feigenbaum is supported in part by NSF grant CNS-1016875 and DARPA contract N66001-11-C-4018.
Jaggard is supported in part by NSF grant CNS-1018557.
Wright is supported in part by NSF grant CNS-1018557.
Xiao is supported by NSF grant CNS-1016875.

With respect to information, we examine the type(s) of credentials the system participants use, what constitutes evidence of compliance with or violation of a security policy, and who must have access to credentials and evidence for the system to function. To what extent does the system rely on participants' identities, and how is "identity" defined? If identity is used, how broadly does a participant's identity become known? Who learns about a violation when one occurs, and how soon after the fact of the violation does he learn it? The role of identity is important because of the widespread but mistaken perception (discussed below) that accountability is inherently in tension with anonymity. Interestingly, some of the works that we cover in this systematization effort regard identification of wrongdoers as the final step in a process—as judgment and punishment, in the terms introduced above. In these systems, an act that violates a security policy triggers the identification of the violator who, until he committed the violation, was anonymous. It is assumed that identification *per se* will ensure that the violator is held accountable, but precisely what it means for someone to be "held accountable" is not specified. At the other end of the spectrum, some of the works that we cover assume that all participants have persistent identities, *i.e.*, that anonymity is not an issue, and deal exclusively with formal protocols for presenting evidence, adjudicating a claimed violation, and meting out punishment if the claim is validated. This lack of agreement about the scope of "accountability" research is one of our main motivations for undertaking this systematization effort.

With respect to action, we examine the operational structures of accountability mechanisms and the systems that use them to achieve privacy and security. Are system actions centralized or decentralized? What actions must be taken to deal with a violation? In particular, does a violation trigger *automatic* punishment (such as the destruction of anonymity discussed above), or must evidence of a violation be presented to a *mediator*, who invokes a formal adjudication protocol and, if necessary, a punishment protocol? If there is a mediator, is it an entity that is already part of the system, or is it someone external to the system (like a judge who is only called in for the purpose of adjudicating)? To what extent does the functioning of the system assume continued participation by or access to the violator? That actions could be tied to consequences automatically, *e.g.*, without identification of the actors or the invocation of a formal adjudication protocol, is not a new or radical idea but rather one that has been the subject of extensive study in at least one discipline, namely Economics, in which the design of *incentive-compatible* systems and protocols is a standard goal. The simplest and best-known example of an incentive-compatible protocol in Economics is the 2^{nd}-price Vickrey auction. The policy that bidders are supposed to comply with is "bid your true value."[1] For many natural distributions on the bidders' values, no bidder can improve his utility by lying; indeed, with positive probability, his utility will be decreased if he lies about his value. Thus, actions are *automatically* tied to consequences, and no explicit punishing action is taken. The violator is not identified, and, in fact, no one else even knows that there was a violation.

One barrier to unification and systematization of this technical area is the word "accountability" itself. In common parlance, "holding him accountable" connotes "making him account for himself" or "making him stand up and be counted." The sentiment conveyed therein has considerable social value, and it causes people to resist using the term to describe approaches that may not entail an official "account" by the wrongdoer. This erroneous assumption that "accountability mechanisms" must require the identification of those who violate policies so that violators can be brought to "account" is widespread in the technical community as well, where it raises the hackles of those who conclude that accountability is inherently in tension with anonymity. The fact that "tying actions to consequences" can be accomplished without identifying wrongdoers, as the study of incentive compatibility in Economics clearly demonstrates, gives us hope that this erroneous assumption can be corrected and that the technical community will embrace accountability as an effective tool in situations where preventive measures are inadequate and will recognize that it does not preclude anonymity.

A. Related work

The focus of this work is on accountability solutions and formalizations in Computer Science; that type of related work is discussed in detail below. Other work on accountability in Computer Science includes arguments by Weitzner *et al*. [1] and by Lampson [2] about the need for accountability and security-by-deterrence (such as might be provided by accountability). In early work on accountability in Computer Science, Nissenbaum [3] studied barriers to accountability in contexts involving software.

Chockler and Halpern [4] build on the Halpern–Pearl [5] framework of causality to give formal definitions of both responsibility (the extent to which something is a cause of an event) and blame (which additionally considers the epistemic state of an agent who causes an event). This does not directly provide a definition of accountability, but these notions might be used to inform actions (such as punishment) taken in response to a policy violation.

Outside of Computer Science, Mulgan [6] has traced the evolution of "accountability" in Public Administration

[1]This might not be explicitly stated as a policy requirement, but that does not affect incentive compatibility. In considering this in the context of accountability, we may assume that we are in a setting in which this goal is an explicit policy and that we want to ensure that violations of this policy are punished.

from its core meaning of being able to be called to give an account (*e.g.*, of one's actions). Grant and Keohane [7] have given a definition in the context of nation states interacting with each other. Our focus is not on approaches outside of computer science, of which these are but a small sample, so we will not discuss them in more detail here. (Feigenbaum *et al*. [8], [9] provide more discussion of non-Computer Science approaches to accountability.)

II. Aspects of Accountability

As we survey approaches to accountability, we evaluate how they address three broad aspects of accountability: time, information, and action. In our analysis, we typically think of accountability with respect to some policy violation (in a very general sense); the "time" aspect considers when the system is invoked relative to the time of the violation. The "information" aspect considers what is known and by whom, while the action aspect concerns what is done and by whom.

A. Time/Goals

In surveying approaches to accountability, it becomes clear that different systems are focused on different times relative to a policy violation; this often corresponds to different goals for the system. As one example, the formal framework of Küsters *et al*. [10] explicitly models (and focuses on) judgments or verdicts, *i.e.*, declarations that a violator is guilty of committing a violation. By contrast, the formal framework of Feigenbaum *et al*. [9] focuses on punishment, which typically follows a declaration of guilt. (Within this punishment-focused framework, there need not be a judgment that identifies an individual entity as guilty; so this is indeed distinct from a focus on judgment.)

Motivated by such differences, we consider a spectrum of times, relative to a policy violation, at which each system/framework/mechanism might play a role. We identify the points below on this spectrum. While we categorize, and refer to, these based in terms of their goals/effects and not in terms of strict temporal relationships, there is a natural temporal ordering of these effects.

Prevention The system is (at least partially concerned) with preventing violations and plays a role before the violation occurs.

Detection The system facilitates, enables, *etc*., detection of a violation (either as it occurs or after it occurs).

Evidence The system helps gather or preserve evidence about a violation that may be used against the accused violator (*e.g.*, in a court of law); in some settings, this may be connected to detection.

Judgment The system renders a verdict about an entity's guilt with respect to a policy violation. (This might be a verdict in a court of law or, *e.g.*, a determination by a system administrator that a particular user violated local policy.)

Punishment The system punishes a policy violator in some way.

As we will observe below, a single system might be involved at multiple points on this spectrum.

B. Information

One question about accountability is the extent to which it implicates privacy. Two aspects of this are the information learned about a violation and the information learned about the violator (or even individuals who do not violate any policy). In studying this, we ask the following related questions about accountability systems:

- Is identity required to participate in the system? If so, how broadly is a participant's identity known (*e.g.*, is it only learned by a trusted third party, is it learned by a limited set of entities, or is it potentially learned by all participants)?
- Are violations disclosed? If so, how broadly (with the same set of possible answers as for identity)? How soon after the violation is this information learned?
- Is the violator identified as such? If so, how broadly is this identification made (with the same set of possible answers as above)?

C. Action

We identify different aspects of actions within the system, both in general operation and to detect and punish policy violations.

- Is the system (as it operates in the absence of a detected violation) centralized or decentralized?
- Does the system respond to a violation (in the gathering of evidence, judgment, and punishment) in a centralized or decentralized way?
- If violators are punished, is this done (in the terms of Feigenbaum *et al*. [9]) "automatically" or in a "mediated" manner? If there is a mediator, is this an entity that is already part of the system, or is it a specialized external entity?
- To what extent does the functioning of the system rely upon continued participation by, or access to, the violator? For example, is the violator only punished if he continues to interact with the system?

D. Applicability of this framework

The three broad aspects described above can be used to characterize various approaches to accountability in Computer Science; we do this in the following section. As new accountability systems and approaches are developed, they can also be analyzed within this framework.

In addition to being broadly applicable, we argue that our framework captures essential aspects of accountability at a useful level of granularity. Insofar as "accountability" relates to violations (of policy, law, *etc*.), either actual or possible, the "time" aspect of our framework allows us to compare the relative times at which different systems have effects. The "information" and "action" aspects separate system

characteristics that should be compared separately without producing an unmanageably high-dimensional framework.

III. Survey of Approaches

There are many different Computer Science approaches to accountability. We discuss a variety of accountability solutions (in Section III-A) and accountability formalizations (in Section III-B) that take on different values along the axes we identified in Section II.

Table I summarizes our analysis of systems and formalizations that exemplify broader areas of accountability research in Computer Science. The columns correspond to the aspects and sub-aspects of accountability discussed in Sec. II; the reasoning that supports the entries in the table is described in the paper section listed in the leftmost column. Where applicable, the discussion in the text also notes other possible answers or answers that might arise in related but distinct solutions or formalizations. Some systems are defined in general ways that do not enforce a particular categorization for some or all of the columns; we discuss below the range of values they allow or what the most likely categorizations are.

A. Accountability solutions

1) PeerReview: The PeerReview system of Haeberlen, Kouznetsov, and Druschel [11] provides a notion of accountability in distributed systems. They take an "accountable" system to be one that "maintains a tamper-evident record that provides non-repudiable evidence of all nodes' actions." In the asynchronous setting considered by Haeberlen *et al*., the possible violations are not responding to a message (to which a response is prescribed by the protocol) or sending a message that is not prescribed by the protocol. The potential for message delays means that the former cannot be conclusively proved; this gives rise to a distinction between suspicion and certainty, both of which are included in the system.[2]

The design of PeerReview includes, at each node in the network, a detector module that implements the system; this will indicate either suspicion or certainty that another node is violating the protocol. It makes use of a tamper-evident log that each node maintains of its own interactions (and that can be obtained by other nodes as they need it). Taken together, these range over the *detection*, *evidence*, and *judgment* parts of the **Information** aspect of our framework.

Nodes must be identified to participate in a distributed protocol that incorporates PeerReview; for the **Information** aspect of our framework, their identity is made known to a *broad* set of other participants. The security goals for PeerReview include that every node that fails to acknowledge a message is eventually suspected of violating the protocol by *every* node that does follow the protocol, so the disclosure of a violation and the identification of the violator as such are *broad/broad*. Under the **Action** aspect, PeerReview is *decentralized* both without and with violations and there is no punishing entity (*not applicable*). If a violator no longer participates, then that node will be viewed as not responding to messages and will be suspected by other nodes; thus, the system *does not* require ongoing involvement on the part of the violator.

2) Anonymous blacklisting systems: Like e-cash systems, anonymous blacklisting systems allow anonymous participation. In contrast to e-cash, participants in these systems are not identified when they commit a violation; instead, they are blacklisted (*i.e.*, their credentials for participation are revoked) without identifying them. Henry and Goldberg have recently surveyed this space of systems [26] and identified three broad subspaces thereof: pseudonym systems, Nymble-like systems, and revocable anonymous credential systems. These provide varying levels of privacy (ranging from pseudonyms to complete anonymity without trusted third parties); however, as the privacy guarantees are strengthened, the feasibility of implementation decreases.

As an exemplar of this class of systems, we will take the PEREA revocable anonymous credential system of Tsang, Au, Kapadia, and Smith [12]. The user must first register with the system. Depending on the setting, this might require some form of identity; however, the user obtains a credential that can subsequently be used to authenticate herself to the service provider without revealing her identity. The service provider may subsequently revoke the client's credential for any reason, without requiring a trusted third party to do so; this prevents the client from authenticating herself in the future, but it does not reveal anything about her identity to anyone (nor does it link her various anonymous actions, among other properties).

For the **Time/Goals** aspect, this provides *punishment (mediated)*, because the punishment is carried out by the service provider (in blacklisting the anonymous credential). While this is presumably based on the detection of some violation and the judgment of guilt, PEREA itself is not used to do these things. For our **Information** aspect of accountability, the system *might* require some sort of identity to register, so we categorize this as *unique*. Importantly, however, the violator is not identified as such (although the violation is known by the service provider), so we categorize the last two sub-aspects of this as *unique/none*. The registration and authentication require some *centralized* aspects (regardless of whether there is a violation); the punishing entity is part of the system (*internal*), but punishment does not require the ongoing participation of the violator (*does not*).

3) Accountable signatures: When digital signatures allow multiple potential signers, either because many individuals could generate the signature or because a valid signature requires multiple signers to generate it, "accountability" has

[2]For example, one system goal is that nodes that ignore messages should eventually be suspected, in perpetuity, by all honest nodes even though they cannot be certain that the ignoring node is in fact ignoring messages.

		Time/Goals					Information			Action			
Section	Approach/Paper	Prevention	Detection	Evidence	Judgment	Punishment	Identity Requirements for Participation	Violation Disclosed?	Violator Identified as Such?	Centralization without Violation?	Centralization with Violation?	Punishing Entity?	Requires Ongoing Involvement?
Solutions													
III-A1	PeerReview [11]		✔	✔	✔		Broad	Broad	Broad	Decent.	Decent.	N/A	No
III-A2	PEREA [12]					Med.	Unique	Unique	No	Cent.	Cent.	Internal	No
III-A3	ASMs [13]	✔	✔	✔			Broad	Broad	Broad	Decent.	Decent.	N/A	No
III-A4	E-Cash [14]		✔	✔	✔		Unique	Broad	Broad	Cent.	Cent.	N/A	No
III-A5	iOwe [15]		✔	✔	✔	Med.	Broad	Broad	Broad	Decent.	Decent.	Internal	No
III-A6	Buchegger & Boudec [16]					Med.	Broad	Broad	Broad	Decent.	Decent.	Internal	Yes
III-A7	A2SOCs [17]			✔	✔	Med.	Unique	Broad	Broad	Cent.	Cent.	Int./Ext.	
Formalizations													
III-B1	Küsters *et al.* [10]		✔	✔	✔		Broad			Decent.	Cent.	N/A	No
III-B2	Bella & Paulson [18]			✔			Limited	Limited	Limited	Cent.	Cent.	N/A	No
III-B3	Yumerefendi & Case [19]–[21]		✔	✔	✔		Broad	Broad	Broad	Cent.	Cent.	N/A	No
III-B4	Feigenbaum *et al.* [9]					A/M							
III-B5	Jagadeesan *et al.* [22]				✔		Broad	Broad	Broad	Decent.	Decent.	N/A	No
III-B5	Barth *et al.* [23]				✔		Broad	Broad	Broad	Cent.	Cent.	N/A	No
III-B6	Kailar [24]			✔			Broad					N/A	
III-B6	Backes *et al.* [25]			✔	✔		Broad					N/A	

Table I
OVERVIEW OF ACCOUNTABILITY APPROACHES.

the potential to become an issue in ways that it is not when there is only one potential signer. There are many different approaches to signatures with multiple potential signers; as an exemplar of this area, we take the work by Micali, Ohta, and Reyzin on "accountable-subgroup multisignatures" [13] that explicitly took "accountability" as a goal. Their definition of this goal was

> Accountability means that, without use of trusted third parties, individual signers can be identified from the signed document.

As noted by Micali *et al.*, other approaches with multiple potential signers allow sets of individuals (possibly just a single individual) to generate signatures on behalf of a larger set of individuals in such a way that the individual(s) who produced the signature cannot be identified.

For accountable-subgroup multisignatures as defined by Micali *et al.*, all members of the group run a key-generation protocol once; the signing protocol takes, from each signer, a description of the set of signers and their public keys, the message being signed, and the individual signer's secret key. The signers then produce the signature, which can be verified (when input with a message and a set of purported signers) by anyone. This is secure (and Micali *et al.* describe a secure scheme for accountable-subgroup multisignatures) if an attacker cannot (except with negligible probability) produce a valid signature for a message m where the set S of individuals who purportedly signed m includes an honest participant who did not execute the signing protocol.[3] The set of purported signers provides a guarantee to the verifier of the signature; the signers may not know each other. As Micali *et al.* note [13]:

> Then, assuming that P_2 [one purported signer] has not been corrupted, P_1 [another purported signer] is assured that the verifier will deem the signature valid only if the person whom the verifier knows

[3]Micali *et al.* [13] discuss issues of adaptive corruption and prove the equivalence of security notions that involve attackers of formally different abilities; those distinctions do not affect our analysis.

as P_2 actually participated in the signing protocol on [m and S].

This approach provides accountability through identity; from the perspective of holding policy violators "accountable," it neither judges nor punishes violators. The **Time/Goals** properties that this approach does provide arguably depend on the type of policy under consideration: the security definition provides *prevention* of successful forgeries and *detection* of forgery attempts, while it provides *evidence* of violations that are carried out by someone using his own identity for signing (analogous to, *e.g.*, an officer of an company signing his/her own name to an improper corporate check). With respect to the **Information** aspects of this approach, identity is definitely requited, and a participant's identity is potentially known to a *broad* set of other individuals. We may take two different views of the questions of whether the violation is disclosed and whether the violator is identified as such. Under the first, the violation consists of an attempted forgery. This is detected by the verifier, but the violator might not be identified; we categorize this case as *unique*/*none*. Under the second view, no forgery is attempted but the (valid) signature on the message indicates that the signers have committed some (non-identity) violation. In this case, the violation is disclosed (because it is embedded in the message), and the violators (the signers) are identified as such, to a broad set of participants; we categorize this as *broad*/*broad*. For the **Action** aspects of this approach, the signatures can be generated in a *decentralized* way (with or without a violation); this does not incorporate punishment, so we consider the punishing entity to be *not applicable*; finally, this *does not* require ongoing involvement by violators.

4) E-cash: Pioneered by David Chaum in the early 1980's [27], [28], e-cash was designed to have the anonymity and untraceability properties of physical cash: A user should be able to withdraw money from the bank and spend it with a merchant without revealing her identity, and the merchant should be able to deposit the money received into his account without the bank learning how individual users spent their money. Due to the replicable nature of the strings of bits that represent digital money, a primary issue to resolve in realizing e-cash is how to prevent or deter "double-spending," in which users or merchants make and spend (or deposit) multiple copies of electronic coins. A solution to this [29] provides consequences for double spending using cryptographic mechanisms that break the anonymity of double spenders. These solutions rely on identity for accountability. Depending on the context of the system, loss of anonymity might or might not be sufficient punishment in and of itself. If not, the system would need to rely on an external mechanism to provide any additional punishment.

Chaum's solutions, and many that grew out of them, have a model in which the bank is a centralized party that checks for double spending. Chaum's initial proposals [27], [28] were "on-line," in the sense that the bank must be involved in every transaction in order to prevent double-spending. Chaum, Fiat, and Naor [29] introduced "off-line" e-cash, in which double-spending was not strictly prevented, but the identity of double-spenders would be revealed by the bank after-the-fact, including providing an incontestable proof of the violation (including protecting against a cheating merchant who might try to collude with a customer in order to undetectably allow double spending and/or attempting to frame an innocent customer as a double-spender).

While a complete survey of e-cash schemes is beyond the scope of this paper, we note that there have been many proposals that take different approaches and provide different properties, including differences in prevention vs. detection, centralization vs. decentralization, and security vs. efficiency. An interesting example in trading off security and efficiency is Rivest and Shamir's MicroMint [30], which is designed so that small-scale fraud will be unprofitable, while large-scale fraud will be detectable.

A recent exemplar of the off-line approach, proposed by Camenisch, Hohenberger, and Lysyanskaya [14], explicitly addresses accountability as a goal to be balanced with privacy, while extending the accountability goals beyond double spending. Specifically, in addition to detection of double spending, their work supports spending limits for each merchant, motivating by concerns that anonymous e-cash can allow undetectable money laundering. A user's anonymity and untraceability is guaranteed as long as she does not violate either policy (double spending or spending limits). Violations can be detected, including determining whether a user or a merchant cheated. When a violation is detected, the bank becomes (mathematically) able to identify the violating user as well as trace the other activities of the violating user. For **Time/Goals**, the system therefore does not rely on prevention. It includes *detection*, *evidence*, and *judgment*. The consequence of detecting cheating is that a user loses her anonymity and untraceability. As noted above, this might be considered to provide sufficient punishment, but in general could need to be supplemented with additional punishment external to the system. Regarding **Information**, identity is an inherent part of the system, but honest parties are guaranteed anonymity. The bank learns about violations and the identity of violators at the time that coins violating the policy are deposited with the bank. For **Action**, the system relies on the bank as a central authority. Users can spend coins at merchants without the involvement of the bank, but users must obtain all coins from the bank and merchants must deposit all coins with the bank, at which time violations can be detected.

5) iOwe: As we have discussed, many e-cash systems rely on the use of a centralized authority in order to provide their security and accountability properties. Given the decentralized context of peer-to-peer systems, it can be undesirable to rely on a centralized authority for monetary

purposes. To this end, a number of decentralized currency systems have been proposed, including [15], [31], [32].

We study the iOwe currency system [15] as an exemplar of such systems. iOwe allows peers in a decentralized peer-to-peer system to exchange currency backed by system resources. Peers create "iotas" as promises of future work. Iotas can be exchanged for work as payment, or "redeemed" with their originators for work, along the line of standard "IOU"s, but with greater liquidity. iOwe does not prevent double-spending, but addresses it by using signature chains that allow detection by a peer (possibly but not necessarily the originator) seeing the same iota twice, using the two signature chains as a proof of misbehavior, and applying a punishment mechanism that expels detected cheaters and all iotas they issued from the system. Thus, on the **Time/Goals** aspect, iOwe uses *detection*, *evidence*, *judgment*, and *punishment (mediated)*.

iOwe peers have a persistent identity within the system, but these identities need not be tied to external identities and users are not prevented from creating multiple peers (or "Sybils") within the system. iOwe limits the potential for a user to benefit from double spending using by adding a layer of reputation to the system. Specifically, peers build up trust of other peers by participating in the system (creating, spending, and redeeming iotas), and peers only accept iotas that were both issued by peers they trust and only ever held by peers they trust. In this way, "Sybil" peers are not able to create iotas, because they have not been able to build up trust. A peer therefore can deflect blame for double-spending to another Sybil node it has created, but the peer will be punished by losing the value of any outstanding iotas it holds that were issued by or ever held by the expelled Sybil. In terms of our **Information** aspect, violators are identified by their (weak) identities within the system. The violation is disclosed to any peer that receives the duplicate iota, possibly when returned to the issuer but possibly earlier. In keeping with the decentralized nature of peer-to-peer systems, the **Action** aspect is entirely decentralized. Both the normal operation and the handling of violations are done in a decentralized way, with individual peers able to detect and verify double spending and to implement their own part of the punishment by no longer trusting the violator. (Similar punishment is extended to peers who refuse to redeem iotas they have issued; we omit discussion of that component of the system here.)

6) Reputation systems: Reputation systems have received much attention in various settings. Even when not explicitly motivated by "accountability," aspects of accountability are closely related to the natural use of these systems. In particular, an action that depends on the reputation of another node could very easily (unless the other node is always indifferent to which action is chosen) be viewed as potential punishment.

As an example of a reputation system, we consider the one for mobile ad-hoc networks proposed by Buchegger and Boudec [16]. Each node i in the network has, for each other node j that it tracks, a trust rating and a reputation rating. The reputation rating, which affects how i behaves towards j, is affected by both i's direct interactions with j and information obtained about j from other nodes (in particular, nodes that either i trusts or that have experiences with j that are similar to i's experiences). If i's view of j is sufficiently bad, then i will avoid routing through j, and i will ignore future route requests from j. (While we view this as punishing j for misbehaving in the routing protocol, Buchegger and Boudec explicitly note that they do *not* punish nodes that give inaccurate reports in the reputation system.) The particular (modified Bayesian) approach to updating reputation is unrelated to the accountability properties of this system.

For the **Time/Goals** aspect of accountability, this provides *punishment (mediated)* through the avoidance of a node in routing and ignorance of its route requests. Arguably, this is also providing a sort of *judgment*, but in an average sense (over many different violations and non-violations); because of that averaging, we will not categorize this as providing *judgment*. (Similarly, this requires that violations are detected, but the reputation system propagates that information instead of actually doing the detecting.) For the **Information** aspect, identity is definitely required[4] and is known to a *broad* set of other participants. Similarly, the point of a reputation system is to identify violators as such (in a fairly broad way), disclosing the violations, so we categorize this as *broad/broad*. For the **Action** aspect, this is *decentralized* both without and with a violation. There are punishing entities—the other nodes in the network, which are *internal*—but punishment *does* rely on the continued participation of the violator (because punishment takes the forms of routing around and ignoring the violator).

7) A2SOCs: Farkas, Ziegler, Meretei, and Lörincz [17] described an approach (Anonymous and Accountable Self-Organizing Communities, or A2SOCs) to "anonymous accountability" with multiple levels of identities (including pseudonyms). They use both "internal" and "external" notions of accountability and give protocols to provide these. The former notion means that a pseudonym can be "held responsible" for its actions (even under different pseudonyms that are not publicly linked); this is done by the other members of the virtual community. By contrast, "external accountability" is used to mean that the real-world entity connected to the pseudonyms is identified and this real-world identity may be given to, *e.g.*, the police when a real-world crime has been committed. Both the linking of different pseudonyms that belong to the same agent and the release of an agent's real-world identity require broad community agreement (although this assumes that the trusted third party

[4]Identity is required to be "persistent, unique, and distinct."

has, as required, deleted keys that it initially used to register pseudonyms).

Within the **Time/Goals** aspect of our framework, we categorize the approach of Farkas *et al*. as providing *evidence* (*e.g.*, through the linking of pseudonyms and providing real-world identities to outside agencies), *judgment* (because the virtual community can, and must, agree to help link different pseudonyms or to reveal a real-world identity), and *punishment (mediated)* (via either the community or the external authorities). Within the **Information** aspect of our framework, identity is initially needed only for registration, which reveals it to the *unique* trusted third party. However, violations are disclosed in a *broad* manner, and (either as a pseudonym or as a real-world identity), violators are identified as such to the *broad* community. The trusted third party means that, in our **Action** aspect, the system is *centralized* both with and without violations. Depending on the level of the violation (and whether pseudonyms are linked or a real-world identity is revealed), the punishing entity can be either internal or external to the community, so we classify this as *both*. The punishment might (*e.g.*, for within-community punishment) or might not (*e.g.*, for banishing a user or for external punishment) require ongoing involvement by the violator, so we do not classify the system in this respect.

B. Formalizations of accountability

There have been several proposed formalizations of accountability. These, too, take different interpretations of accountability and therefore can apply to different solutions or to different properties of those solutions. We discuss different approaches to formalizing accountability, as well as one that formalizes the related notion of auditability.

1) Accountability through judging: Küsters, Truderung, and Vogt provide a model for accountability. In an intuitive description of their definitions, a protocol provides accountability if a specified "judge" (who might or might not have an additional role in the protocol) is able to issue "verdicts" about misbehaving participants ("violators," in our terminology) in a way that is both fair and complete. Specifically, the judge should never blame protocol participants who behave honestly (fairness), and, whenever the protocol fails to meet its specified goals, the judge should blame at least some misbehaving participants (completeness). It is left external to this analysis what the consequences for violators should be and how they should be enforced. Thus, for our **Time/Goals** aspect, the model addresses *detection*, *evidence*, and *judgment*. Note that the judge is not required to produce evidence in the form of proofs that others can use, but, if the system provably satisfies fairness, the very existence of the judge's verdict in fact serves as that evidence.

The required verdicts in their model are positive Boolean formulae "built from propositions of the form $\mathsf{dis}(a)$, for an agent a, where $\mathsf{dis}(a)$ is intended to express that a misbehaved." Thus, for our **Information** aspect, this method relies on participating agents to have identities in whatever system is being analyzed. They do, however, allow for the possibility of verdicts that do not identify individual violators, by allowing disjuncts. In this sense, a violator might or might not be explicitly identified as such. (However, they point out that individual accountability, in which individual violators are identified, is highly desirable in practice to deter parties from misbehaving.) Violations are disclosed at least to the judge, as well as to any parties to whom the judge shares the verdict.

For **Action**, the definition requires the existence of a judge to provide the required verdicts, suggesting a centralized system. However, one could imagine applying their definitions to a decentralized system such as iOwe [15], described in Sec. III-A5, where different parties can act as judges for different violations, for example, proving results such as: if party P double-spends an iota, then another party Q can act as a judge and hold P accountable.

2) Connecting actions to identities: Bella and Paulson [18] have formalized properties of two particular "accountability protocols" and verified these using the Isabelle tool; these protocols connect actions to identities.[5] The particular protocols that they studied were for non-repudiation [34] and certified email [35]; here, we focus not on these protocols individually but on the class of accountability protocols that they exemplify (*i.e.*, that corresponds to the properties identified in [18]).

In the approach of Bella and Paulson,

> [a]n accountability protocol gives agents lasting evidence, typically digitally signed, about actions performed by his peer.

They note that many authentication protocols prove the involvement of a participant to one other participant (*e.g.*, via an encrypted nonce), but that these do not provide evidence that is suitable to take to a third party. Indeed, one of the two goals they identify for accountability protocols is that

> an agent is given evidence sufficient to convince a third party of his peers participation in the protocol.

(The other goal is a notion of fairness in which either both participants receive, or neither participant receives, evidence about the other's participation.) Bella and Paulson explicitly note that judging is left to humans who are not modeled in their analysis. For the **Time/Goals** aspect of our framework, we thus say that their approach (and the accountability solutions that fall within their model) provide *evidence* but not other parts of this aspect.

Both protocols considered by Bella and Paulson involve two regular participants a trusted third party; all three of

[5]This broad approach is also embodied in the Accountable Internet Protocol [33] of Andersen *et al*. They identify the lack of accountability with the fact that "the Internet architecture has no fundamental ability to associate an action with the responsible entity."

these parties learn the identities of the participants, but those identities are not broadcast further. For the **Information** aspect of our framework, we will thus say that identity is required in a *limited* sense. Violations (captured in the protocol exchanges, not attempts to circumvent the protocols themselves) are revealed through the evidence that the protocols provide, and the violators are identified as such; because this information is provided to the other participant but not broadcast, we say that the other two parts of this aspect are *limited* and *limited*.

For the **Action** aspect of our framework, the trusted third party is required regardless of whether there is a violation, so we identify the Bella–Paulson approach as *centralized*/*centralized*. There is no punishing entity (*not applicable*), and there is no requirement that a violator continue to participate in the protocol (*does not*).

3) Accountability for network services: Yumerefendi and Chase [19] have outlined an approach to accountability for network services that respond to client requests. In so doing, they have articulated a definition of accountable services, described a general method for achieving this, and sketched its application to three different settings; they subsequently used this approach in a more detailed description of network storage with accountability [21]. Here, we are interested in their general definition and method, which may be applied beyond the settings they noted.

Yumerefendi and Chase say that accountable systems should have actions that are: provable and non-repudiable; verifiable (with the ability to prove misbehavior to any third party); and tamper-evident (regarding the states of the system). (These foster the goal they identified in related work [20], which argued for accountability as a design goal, of "assign[ing] responsibility for states and actions[.]") Considering the **Time/Goals** aspect of our framework, this means that the systems provide both *detection* and *evidence*. It is envisioned that auditors are involved (who might examine the evidence that the service has behaved correctly); while these might arguably be viewed as lying outside of this system, we will include them (clients may verify that the service correctly maintained its state) and so also view this approach as providing *judgment*. (The subsequent extension of this approach to network storage [21] reinforces audit as an important component of this approach and the fact that any participant may act as an auditor.)

This approach to accountability has systems publish signed, non-repudiable digests of their internal states. As Yumerefendi and Chase observe, client actions (and potentially identities) may be incorporated into the services' states, so, under our **Information** aspect, we categorize this as requiring identity that may be revealed (or at least checkable by) a *broad* set of participants. Violations are also identified to a *broad* set, and violators are likewise identified to a *broad* set of participants.

For the **Action** aspect of our framework, regardless of whether there is a violation, the service plays a central role in providing digests and proofs of its correct behavior, so we identify this as *centralized*/*centralized*. There is no punishing entity (*not applicable*). Although the service publishes digests of its state, it needs to respond to later requests to provide proofs of its correct behavior. Insofar as the system is aiming to provide proofs of correct behavior, this requires ongoing participation; however, if others will be at least suspicious if no proof is forthcoming, then this *does not* require the ongoing participation of the violator.

4) Accountability in terms of punishment: Feigenbaum, Jaggard, and Wright [9] give an abstract definition of accountability in terms of punishment and then capture this formally in terms of traces and utility functions. Their definition of accountability includes punishment that is "automatic" in the sense that it is not meted out in conscious response to a violation (which would be "mediated" punishment as noted above). Coupled with this, they also explicitly do not require identity, and they note the possibility of punishment occurring (thus providing accountability) without anyone other than the violator knowing that a violation occurred.

The Feigenbaum *et al*. framework can capture systems with, *e.g.*, varying identity requirements, so the classifications that we have been using for the **Information** and **Action** aspects of accountability are not at all determined without considering a particular system. For the **Time/Goals** aspect, this framework addresses only *punishment (automatic and mediated)* and no other sub-aspects.

5) Accountability through audit: As one exemplar of accountability through the use of auditing, we consider the work of Jagadeesan, Jeffrey, Pitcher, and Riely [22], who describe a formal operational model for distributed systems with a notion of accountability that is obtained through auditing. In particular, the auditor(s) in a system may "blame" a set of participants for a violation, *i.e.*, name the members of that set as potential violators. This gives rise to multiple desiderata (such as whether everyone blamed is a violator and whether all non-violators are able to ensure that they are not blamed) for the audit system; these are treated as accountability properties, but they do not change the underlying approach of blaming (sets of) individuals for violations.

We identify the blaming of individuals in the Jagadeesan *et al*. model with *judgment* within the **Time/Goals** aspect of our framework. (While the auditors rely upon evidence to make their judgments, the notion of accountability captured by this framework seems to fit much more with the judgment itself.) Because sets of individuals are blamed using their identities, some sort of identity is required to participate; while this might not be broadcast throughout the system, there are no restrictions on it, so, within the **Information** aspect, we say that the identity required to participate is *broad*. Violations are disclosed, and violators are identified as such, in similar ways, so we identify those parts of this

aspect as *broad/broad*. While auditors are trusted in this system, they do not have a global view (*i.e.*, they interact with the system as participants); for the **Action** aspect of accountability, we thus say that the centralization without/with a violation is *decentralized/decentralized*. This framework is not concerned with punishment, so the punishing entity is *not applicable*. Judgment can be made without the presence of the violator, so this system *does not* require the ongoing participation of a violator.

As a second exemplar of accountability through audit, we note the work of Barth, Datta, Mitchell, and Sundaram [23], who defined a logic for utility and privacy that they applied to models of business practices (such as healthcare systems). In their application to healthcare systems, agents in the system are responsible for things like tagging messages (*e.g.*, to ensure that sensitive health information is not forwarded to the agents responsible for scheduling patient appointments). Barth *et al*. say that an agent is accountable for a policy violation if the agent did an action that occurred before the violation (from some perspective on the system's behavior) and also did not fulfil his responsibilities. They then give an algorithm to identify accountable agents (via communication logs). While an "accountable agent" might not be the cause of the violation in question, this can be determined by a human auditor, who can repeat the process until the agent who caused the violation is identified.

Within the **Time/Goals** aspect of our framework, this approach has elements of *detection*, *evidence*, and *judgment*. (The last is as with the work of Jagadeesan *et al*.; here we also include the first two elements because *evidence* is provided through the tagging requirements, and *detection* is provided by the notion of "suspicious" events, which can be used to find incorrectly tagged messages.) With respect to identities, the disclosure of violations, and the identification of violators, this approach is similar to that of Jagadeesan *et al*.; in the **Information** aspect of our system, we thus say that the approach of Barth *et al*. is *broad/broad/broad*. Here, the auditing engine (which is used even in the absence of a violation) and the human auditors (who determines whether an agent is the cause of a violation) appear to be centralized, so we say that, in the **Action** aspect, centralization without/with a violation is again *centralized/centralized*. Similarly to the Jagadeesan *et al*. approach, the punishing entity is *not applicable*, and the system *does not* require the ongoing participation of a violator.

6) Analyzing accountability in logical frameworks: Kailar [24] developed a logical framework for analyzing accountability in communication protocols and considered sample applications to electronic-commerce protocols. He defined accountability as

> the property whereby the association of a unique originator with an object or action can be proved to a third party (*i.e.*, a party who is different from the originator and the prover).

Accountability goals in a protocol might include that a customer can prove that a business agreed to sell a particular item at a particular price or that the business can prove it provided that item to the customer. Once these goals are formalized for a particular protocol, and the message contents are formalized, Kailar's framework can be used to derive information about who can prove what to whom. These results can then be compared against the original accountability goals.

Within the **Time/Goals** aspect of our framework, we categorize Kailar's approach as providing *evidence* because the analysis of a particular protocol can determine whether an association between an agent and an action/object can be proved to a third party (although it is the underlying protocol itself that actually provides the evidence). Considering the **Information** aspect of our framework, the use of identities are inherent in Kailar's definition of accountability. It is most natural for these identities to become broadly known through participation in a protocol (*e.g.*, when signing messages that might be seen by any agent on the network), and there is no restriction on their distribution built in to the logical framework, so we classify this as *broad*. The disclosure of violations and the identification of violators as such might vary across protocols analyzed using Kailar's framework, so we do not classify it in these respects. Similarly, the different components of the **Action** aspect of our framework are not relevant at this level of abstraction, so we do not classify Kailar's framework with respect to those.

Backes, Datta, Derek, Mitchell, and Turuani [25] used a protocol logic (similar to one originally used for authentication) to prove properties of contract-signing protocols. One of these properties was accountability, which they defined as follows:

> Accountability means that if one of the parties gets cheated as a result of [the trusted third party] $\hat{T}$'s misbehavior, that it will be able to hold $\hat{T}$ accountable. More precisely, at the end of every run where an agent gets cheated, its trace together with a contract of the other party should provide non-repudiable evidence that $\hat{T}$ misbehaved.

As an example of such evidence, Backes *et al*. give the example of terms that can be used (in the logic they define) to derive a term that captures the dishonesty of the trusted third party.

Considering the approach of Backes *et al*. within the **Time/Goals** aspect of our framework, we say that this is focused on determining whether the protocols they study provide *evidence*. (Because this is defined in terms of being able to derive a judgment of dishonesty using the protocol logic, this also has aspects of *judgment*.) Within the **Information** aspect of our framework, we say that this requires *broad* knowledge of identity (because this concerns

the behavior of trusted third parties); the disclosure of the violation and the identification of the violator as such are not determined by the Backes *et al*. framework (although these would likely be *broad*). The presence of the trusted third party means that the contract-signing protocols have a centralized aspect to them, but this requirement is not imposed on the accountability analysis (although it seems that the proof of dishonesty would typically be derived without centralization). There is no punishing entity, and ongoing involvement by the dishonest trusted third party is also not determined.

IV. Conclusions

Our systematic consideration of many major works on "accountability" demonstrates conclusively that computer scientists have used the term to mean different things. We have organized prior work on accountability along the axes of time, information, and actions and highlighted both existing results and open questions. Interestingly, our decision to define accountability mechanisms as those that allow actions to be tied to consequences (and, in particular, allow violations to be tied to punishment) dispels the mistaken notions that accountability precludes anonymity and that it requires centralized authority.

Our systematization effort has revealed the need for more sparing use of the word "accountability" and, more generally, for more precise and consistent terminology. In particular, destroying the anonymity of the violator of a security policy is more accurately described as "identification" or "exposure" than as "accountability." Consistent and more focused use of the term "accountability" should promote the formation of a coherent research area and the adoption of the technology that it develops.

As discussed elsewhere [9], one challenge in addressing punishment is separating punishment for a violation from other, unrelated events that might occur between the violation and the punishment. Other challenges (especially in implementing systems for accountability) include calibrating the severity of the punishment so that it is an effective deterrent (despite the fact that different participants may view the cost of a particular punishment very differently) and determining how often punishment should be meted out.[6] In addition to these punishment-related issues, our work here highlights and distinguishes differing approaches to the detection–evidence–judgment–punishment spectrum and to questions of information and action. These different approaches will inform further analysis of accountability, including the study of fundamental tradeoffs related to accountability, and the design of new accountability systems.

While we have focused on accountability in Computer Science here, the aspects of accountability that we use in our analysis might also be applied to accountability in other disciplines (*e.g.*, the notion of "calling to account" within a particular legal or political framework). This work might thus facilitate comparisons and interactions between notions of accountability in different disciplines.

Finally, we remark that the work we have presented herein is about accountability with respect to established policies. Yet, there are forms of online life, including search and social networking, in which expectations, laws, and policies are still developing. Despite the fact that their obligations have not yet been fully formalized and are not yet fully agreed upon, it would be highly desirable to be able to hold the companies that provide search, social networking, and other online services accountable if, at some point in the future, they are seen to have acted egregiously. As work on accountability in computer science continues, this issue should receive more attention.

[6] There are certainly occasions on which punishment might be withheld in order to promote some larger goal, but if punishment is *always* withheld, the system would not provide accountability.

References

[1] D. J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigenbaum, J. Hendler, and G. J. Sussman, "Information accountability," *Commun. ACM*, vol. 51, pp. 82–87, Jun. 2008. http://doi.acm.org/10.1145/1349026.1349043

[2] B. Lampson, "Privacy and security: Usable security: how to get it," *Commun. ACM*, vol. 52, pp. 25–27, November 2009. http://doi.acm.org/10.1145/1592761.1592773

[3] H. Nissenbaum, "Accountability in a computerized society," in *Human values and the design of computer technology*, B. Friedman, Ed. Stanford, CA, USA: Center for the Study of Language and Information, 1997, pp. 41–64.

[4] H. Chockler and J. Y. Halpern, "Responsibility and blame: A structural-model approach," *Journal of Artificial Intelligence Research*, vol. 22, pp. 93–115, 2004. http://dx.doi.org/10.1613/jair.1391

[5] J. Y. Halpern and J. Pearl, "Causes and explanations: A structural-model approach–part I: Causes," *British J. Philos. Sci.*, vol. 56, pp. 843–887, 2005.

[6] R. Mulgan, "'Accountability': An ever-expanding concept?" *Public Administration*, vol. 78, no. 3, pp. 555–573, 2000.

[7] R. W. Grant and R. O. Keohane, "Accountability and abuses of power in world politics," *American Political Science Review*, vol. 99, no. 01, pp. 29–43, 2005.

[8] J. Feigenbaum, J. A. Hendler, A. D. Jaggard, D. J. Weitzner, and R. N. Wright, "Accountability and deterrence in online life (extended abstract)," in *ACM Web Science*, 2011, http://www.websci11.org/fileadmin/websci/Papers/35_paper.pdf.

[9] J. Feigenbaum, A. D. Jaggard, and R. N. Wright, "Towards a formal model of accountability," in *Proceedings of NSPW '11*. New York, NY, USA: ACM, 2011, pp. 45–56. http://doi.acm.org/10.1145/2073276.2073282

[10] R. Küsters, T. Truderung, and A. Vogt, "Accountability: definition and relationship to verifiability," in *Proceedings of ACM CCS '10*. New York, NY, USA: ACM, 2010, pp. 526–535. http://doi.acm.org/10.1145/1866307.1866366

[11] A. Haeberlen, P. Kouznetsov, and P. Druschel, "Peerreview: practical accountability for distributed systems," *SIGOPS Oper. Syst. Rev.*, vol. 41, pp. 175–188, October 2007. http://doi.acm.org/10.1145/1323293.1294279

[12] P. P. Tsang, M. H. Au, A. Kapadia, and S. W. Smith, "PEREA: towards practical TTP-free revocation in anonymous authentication," in *Proceedings of ACM CCS '08*. New York, NY, USA: ACM, 2008, pp. 333–344. http://doi.acm.org/10.1145/1455770.1455813

[13] S. Micali, K. Ohta, and L. Reyzin, "Accountable-subgroup multisignatures: extended abstract," in *Proceedings of ACM CCS '01*. New York, NY, USA: ACM, 2001, pp. 245–254. http://doi.acm.org/10.1145/501983.502017

[14] J. Camenisch, S. Hohenberger, and A. Lysyanskaya, "Balancing accountability and privacy using e-cash (extended abstract)," in *SCN*, 2006, pp. 141–155.

[15] D. Levin, A. Schulman, K. Lacurts, N. Spring, and B. Bhattacharjee, "Making currency inexpensive with iOwe," in *Proceedings of NetEcon'11*, 2011.

[16] S. Buchegger and J.-Y. Le Boudec, "A robust reputation system for mobile ad-hoc networks," Tech. Rep.

[17] C. Farkas, G. Ziegler, A. Meretei, and A. Lörincz, "Anonymity and accountability in self-organizing electronic communities," in *Proceedings of ACM WPES '02*. New York, NY, USA: ACM, 2002, pp. 81–90. http://doi.acm.org/10.1145/644527.644536

[18] G. Bella and L. C. Paulson, "Accountability protocols: Formalized and verified," *ACM Trans. Inf. Syst. Secur.*, vol. 9, pp. 138–161, May 2006. http://doi.acm.org/10.1145/1151414.1151416

[19] A. R. Yumerefendi and J. S. Chase, "Trust but verify: accountability for network services," in *Proceedings of the 11th ACM SIGOPS European workshop*. New York, NY, USA: ACM, 2004. http://doi.acm.org/10.1145/1133572.1133585

[20] ——, "The role of accountability in dependable distributed systems," in *Proceedings of HotDep'05*. Berkeley, CA, USA: USENIX Association, 2005, pp. 3–3. http://dl.acm.org/citation.cfm?id=1973400.1973403

[21] ——, "Strong accountability for network storage," *Trans. Storage*, vol. 3, October 2007. http://doi.acm.org/10.1145/1288783.1288786

[22] R. Jagadeesan, A. Jeffrey, C. Pitcher, and J. Riely, "Towards a theory of accountability and audit," in *Proceedings of ESORICS'09*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 152–167. http://portal.acm.org/citation.cfm?id=1813084.1813098

[23] A. Barth, A. Datta, J. Mitchell, and S. Sundaram, "Privacy and utility in business processes," in *Proceedings of IEEE CSF'07*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 279–294. http://dl.acm.org/citation.cfm?id=1270382.1270658

[24] R. Kailar, "Accountability in electronic commerce protocols," *IEEE Trans. Softw. Eng.*, vol. 22, pp. 313–328, May 1996. http://dl.acm.org/citation.cfm?id=234708.234712

[25] M. Backes, A. Datta, A. Derek, J. C. Mitchell, and M. Turuani, "Compositional analysis of contract-signing protocols," *Theor. Comput. Sci.*, vol. 367, pp. 33–56, November 2006. http://dl.acm.org/citation.cfm?id=1226644.1226647

[26] R. Henry and I. Goldberg, "Formalizing anonymous blacklisting systems," in *Proceedings of the 2011 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 2011, pp. 81–95. http://dx.doi.org/10.1109/SP.2011.13

[27] D. Chaum, "Blind signatures for untraceable payments," in *CRYPTO*, 1982, pp. 199–203.

[28] ——, "Blind signature system," in *CRYPTO*, 1983, p. 153.

[29] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," in *Advances in Cryptology*, (CRYPTO '88). New York, NY, USA: Springer-Verlag New York, Inc., 1990, pp. 319–327. http://dl.acm.org/citation.cfm?id=88314.88969

[30] R. L. Rivest and A. Shamir, "Payword and micromint: Two simple micropayment schemes," in *Proceedings of the International Workshop on Security Protocols*. London, UK: Springer-Verlag, 1997, pp. 69–87. http://dl.acm.org/citation.cfm?id=647214.720369

[31] S. Zhong, J. Chen, and Y. R. Richard Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proceedings of IEEE INFOCOM'03*.

[32] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," http://bitcoin.org/bitcoin.pdf.

[33] D. G. Andersen, H. Balakrishnan, N. Feamster, T. Koponen, D. Moon, and S. Shenker, "Accountable internet protocol (AIP)," *SIGCOMM Comput. Commun. Rev.*, vol. 38, pp. 339–350, August 2008. http://doi.acm.org/10.1145/1402946.1402997

[34] J. Zhou and D. Gollman, "A fair non-repudiation protocol," in *Proceedings of the 1996 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 1996, pp. 55–. http://portal.acm.org/citation.cfm?id=525080.884255

[35] M. Abadi, N. Glew, B. Horne, and B. Pinkas, "Certified email with a light on-line trusted third party: design and implementation," in *Proceedings of WWW '02*. New York, NY, USA: ACM, 2002, pp. 387–395. http://doi.acm.org/10.1145/511446.511497