Reasoning About Delegation and Account Access in Retail Payment Systems

Shiu-Kai Chin and Susan Older

EECS Department, Syracuse University, Syracuse, New York 13244, USA

Abstract. Delegation and trust are essential to the smooth operation of large, geographically distributed systems, such as the US electronic retail payment system. This system supports billions of electronic transactions—from routine banking and store purchases to electronic commerce on the Internet. Because such systems provide the electronic fabric of our networked information society, it is crucial to understand rigorously and precisely the basis for the delegation and trust relationships in them. In this paper, we use a modal logic for access control to analyze these relationships in the context of checks (and their electronic equivalents) as payment instruments. While not free from risk, the retail payment system effectively balances trust, delegation, and risk on billions of transactions. Our logic allows us to explore with rigor the details of trust, delegation, and risk in these transactions.

Keywords: Access control, delegation, trust, retail payment systems, modal logic.

1 Introduction

> *You may be deceived if you trust too much, but you will live in torment if you don't trust enough.*—Frank Crane

Trust—by which we mean the willingness to adopt someone else's beliefs as one's own—is central to the operation of the electronic retail payment system in the US. Trusted delegates operate on behalf of consumers, banks, and financial networks throughout the retail payment system, which handles many billions of transactions yearly.

Systems such as the retail payment system exemplify critical systems that are large, geographically distributed, part of critical infrastructure, and widely used. The electronic retail payment system in particular uses delegation extensively and depends on trust relationships each and every step of the way. Systems engineers who build such systems are ultimately responsible for assuring that the system behaves securely—in this case allowing account access to only those who should have access. Providing this assurance means engineers need formal tools to describe and analyze trust, delegation, and access policies to derive and justify access-control decisions. These tools ideally are both simple and effective. An engineering artifact will often produce an effect that is not precisely understood and demands scientific analysis. In the case of large distributed systems, the retail payment system

V. Gorodetsky, I. Kotenko, and V.A. Skormin (Eds.): MMM-ACNS 2007, CCIS 1, pp. 99–114, 2007.

largely works in a trustworthy fashion. Our objective is to describe and analyze the retail payment system so that we know precisely why it works.

In this paper, we incorporate a formal accounting of delegation into a modal logic for access control, described initially in [1,2] and based on the work of Lampson, Abadi, Burrows and colleagues [3,4]. The extension itself is quite simple, but it seems to work well for analyzing fairly complicated systems. To demonstrate its suitability for use by engineers, we use the logic to formally explicate the policies, delegations, and trust assumptions on which the United States' Automated Clearing House (ACH) network [5] depends. The ACH network is used by depository financial institutions (e.g., banks) to settle large numbers of financial transactions on a daily basis. These transactions are handled electronically according to the rules of the ACH network, using *check truncation*: the physical checks are removed from the payment-processing process, and instead, information from checks is transmitted electronically via the ACH network. The original checks are usually destroyed shortly after they are received by banks. As mandated by the Check 21 Act [6], the substitute electronic checks have the same legal standing as the original paper checks.

The remainder of this paper is organized as follows. In Section 2 we describe our logic for reasoning about access control and introduce our extension for delegation. Section 3 presents how electronic checks are used in the ACH network. We conclude in Section 4.

2 A Logic for Reasoning About Access Control

To keep this paper self-contained, we provide a brief introduction to the access-control logic described in detail in [2,1]. In Section 2.4, we introduce a minor extension to capture delegation and its essential properties.

2.1 Overview of the Logic

Principal Expressions. We start out by introducing a collection of principal expressions, ranged over by P and Q. Letting A range over a countable set of simple principal names, the abstract syntax of principal expressions is given as follows:

$$P ::= A \ / \ P\&Q \ / \ P \mid Q$$

The principal $P\&Q$ ("P in conjunction with Q") represents an abstract principal who makes exactly those statements made by both P and Q; $P \mid Q$ ("P quoting Q") represents an abstract principal corresponding to principal P quoting principal Q.

Access Control Statements. The abstract syntax of statements (ranged over by φ) is defined as follows, where P and Q range over principal expressions and p ranges over a countable set of *propositional variables*:

$$\varphi ::= p \;/\; \neg\varphi \;/\; \varphi_1 \wedge \varphi_2 \;/\; \varphi_1 \vee \varphi_2 \;/\; \varphi_1 \supset \varphi_2 \;/\; \varphi_1 \equiv \varphi_2 \;/\; P \Rightarrow Q \;/\; P \;\mathsf{says}\; \varphi \;/\; P \;\mathsf{controls}\; \varphi \;/\; P \;\mathsf{reps}\; Q \;\mathsf{on}\; \varphi$$

Informally, a formula $P \Rightarrow Q$ (pronounced "P speaks for Q") indicates that *every* statement made by P can also be viewed as a statement from Q. A formula P controls φ is syntactic sugar for the implication $(P \text{ says } \varphi) \supset \varphi$: in effect, P is a trusted authority with respect to the statement φ. P reps Q on φ denotes that P is Q's delegate on φ; it is syntactic sugar for $(P \text{ says } (Q \text{ says } \varphi)) \supset Q \text{ says } \varphi$. Notice that the definition of P reps Q on φ is a special case of controls and in effect asserts that P is a trusted authority with respect to Q saying φ.

2.2 Semantics

The semantics of formulas is based on Kripke structures, as given by the following definitions.

Definition 1. *A* Kripke structure $\mathcal{M}$ *is a three-tuple* $\langle W, I, J\rangle$*, where:*

- W *is a nonempty set, whose elements are called* worlds*.*
- $I : \boldsymbol{PropVar} \rightarrow \mathcal{P}(W)$ *is an* interpretation *function that maps each propositional variable* p *to a set of worlds.*
- $J : \boldsymbol{PName} \rightarrow \mathcal{P}(W \times W)$ *is a function that maps each principal name* A *to a relation on worlds (i.e., a subset of* $W \times W$*).*

We extend J to work over arbitrary *principal expressions* using set union and relational composition as follows:

$$J(P\&Q) = J(P) \cup J(Q)$$
$$J(P \mid Q) = J(P) \circ J(Q),$$

where

$$J(P) \circ J(Q) = \{(w_1, w_2) \mid \exists w'.(w_1, w') \in J(P) \text{ and } (w', w_2) \in J(Q)\}$$

Definition 2. *Each Kripke structure* $\mathcal{M} = \langle W, I, J\rangle$ *gives rise to a function*

$$\mathcal{E}_{\mathcal{M}}[\![-]\!] : \boldsymbol{Form} \rightarrow \mathcal{P}(W),$$

where $\mathcal{E}_{\mathcal{M}}[\![\varphi]\!]$ *is the set of worlds in which* φ *is considered true.* $\mathcal{E}_{\mathcal{M}}[\![\varphi]\!]$ *is defined inductively on the structure of* φ*, as shown in Figure 1.*

Note that, in the definition of $\mathcal{E}_{\mathcal{M}}[\![P \text{ says } \varphi]\!]$*,* $J(P)(w)$ *is simply the image of world* w *under the relation* $J(P)$*.*

2.3 Inference Rules

The semantic functions $\mathcal{E}_{\mathcal{M}}$ provide a fully defined and fully disclosed interpretation for the formulas of the logic. This mathematical foundation enables us

$$
\begin{aligned}
\mathcal{E}_{\mathcal{M}}[\![p]\!] &= I(p) \\
\mathcal{E}_{\mathcal{M}}[\![\neg\varphi]\!] &= W - \mathcal{E}_{\mathcal{M}}[\![\varphi]\!] \\
\mathcal{E}_{\mathcal{M}}[\![\varphi_1 \wedge \varphi_2]\!] &= \mathcal{E}_{\mathcal{M}}[\![\varphi_1]\!] \cap \mathcal{E}_{\mathcal{M}}[\![\varphi_2]\!] \\
\mathcal{E}_{\mathcal{M}}[\![\varphi_1 \vee \varphi_2]\!] &= \mathcal{E}_{\mathcal{M}}[\![\varphi_1]\!] \cup \mathcal{E}_{\mathcal{M}}[\![\varphi_2]\!] \\
\mathcal{E}_{\mathcal{M}}[\![\varphi_1 \supset \varphi_2]\!] &= (W - \mathcal{E}_{\mathcal{M}}[\![\varphi_1]\!]) \cup \mathcal{E}_{\mathcal{M}}[\![\varphi_2]\!] \\
\mathcal{E}_{\mathcal{M}}[\![\varphi_1 \equiv \varphi_2]\!] &= \mathcal{E}_{\mathcal{M}}[\![\varphi_1 \supset \varphi_2]\!] \cap \mathcal{E}_{\mathcal{M}}[\![\varphi_2 \supset \varphi_1]\!] \\
\mathcal{E}_{\mathcal{M}}[\![P \Rightarrow Q]\!] &= \begin{cases} W, & \text{if } J(Q) \subseteq J(P) \\ \emptyset, & \text{otherwise} \end{cases} \\
\mathcal{E}_{\mathcal{M}}[\![P \text{ says } \varphi]\!] &= \{w | J(P)(w) \subseteq \mathcal{E}_{\mathcal{M}}[\![\varphi]\!]\} \\
\mathcal{E}_{\mathcal{M}}[\![P \text{ controls } \varphi]\!] &= \mathcal{E}_{\mathcal{M}}[\![(P \text{ says } \varphi) \supset \varphi]\!] \\
\mathcal{E}_{\mathcal{M}}[\![P \text{ reps } Q \text{ on } \varphi]\!] &= \mathcal{E}_{\mathcal{M}}[\![P \mid Q \text{ says } \varphi \supset Q \text{ says } \varphi]\!]
\end{aligned}
$$

Fig. 1. Semantics

to provide a means to reason about access-control situations using a small core collection of sound inference rules and syntactic-sugar definitions (see Figure 2), along with a larger set of rules that can be formally derived from the core rules (see Figure 3 for a sample set of derived rules that we have found particularly useful).

A rule of form $\dfrac{H_1 \cdots H_n}{C}$ is sound provided that, *for all Kripke structures* $\mathcal{M} = \langle W, I, J \rangle$, *if* $\mathcal{E}_{\mathcal{M}}[\![H_i]\!] = W$ *for each* $i \in \{1, \ldots, n\}$, *then* $\mathcal{E}_{\mathcal{M}}[\![C]\!] = W$. The rules in Figures 2 and 3 are all sound, and become the basis for reasoning about access-control decisions. The Kripke structures are then only necessary if one wishes to add new inference rules and to verify their soundness.

2.4 Delegation and Its Properties

Delegation is an important relationship in networks where decisions and authorities are distributed in different locations. When principal P acts on behalf of principal Q, we say that P is Q's delegate. P acting as Q's delegate on the statement φ, denoted by $P \text{ reps } Q \text{ on } \varphi$, is defined as syntactic sugar:

$$P \text{ reps } Q \text{ on } \varphi \stackrel{\text{def}}{=} (P \text{ says } (Q \text{ says } \varphi)) \supset Q \text{ says } \varphi$$

Essentially, if P is Q's representative on statements φ, then P claiming Q has said φ is treated as if Q said φ herself.

There are three crucial properties of the delegation relationship that our logic (or any other formal system) must accurately capture:

1. A recognized delegate should in fact have the authority to act on behalf of the principals they represent. That is, if a given policy allows principals to delegate to others and recognizes that Bob is Alice's delegate, then Bob should be able to act on Alice's behalf.

$$\textit{Taut}\ \frac{}{\varphi}\quad \text{if } \varphi \text{ is an instance of a prop-logic tautology}$$

$$\textit{Modus Ponens}\ \frac{\varphi \quad \varphi \supset \varphi'}{\varphi'} \qquad \textit{Says}\ \frac{\varphi}{P\ \mathsf{says}\ \varphi}$$

$$\textit{MP Says}\ \frac{}{(P\ \mathsf{says}\ (\varphi \supset \varphi')) \supset (P\ \mathsf{says}\ \varphi \supset P\ \mathsf{says}\ \varphi')}$$

$$\textit{Speaks For}\ \frac{}{P \Rightarrow Q \supset (P\ \mathsf{says}\ \varphi \supset Q\ \mathsf{says}\ \varphi)}$$

$$\textit{Quoting}\ \frac{}{P \mid Q\ \mathsf{says}\ \varphi \equiv P\ \mathsf{says}\ Q\ \mathsf{says}\ \varphi}$$

$$\&\textit{Says}\ \frac{}{P\&Q\ \mathsf{says}\ \varphi \equiv P\ \mathsf{says}\ \varphi \wedge Q\ \mathsf{says}\ \varphi}$$

$$\textit{Idempotency of} \Rightarrow\ \frac{}{P \Rightarrow P} \qquad \textit{Monotonicity of} \mid\ \frac{P' \Rightarrow P \quad Q' \Rightarrow Q}{P' \mid Q' \Rightarrow P \mid Q}$$

$$\textit{Associativity of} \mid\ \frac{P \mid (Q \mid R)\ \mathsf{says}\ \varphi}{(P \mid Q) \mid R\ \mathsf{says}\ \varphi}$$

$$P\ \mathsf{controls}\ \varphi \quad \stackrel{\text{def}}{=} \quad (P\ \mathsf{says}\ \varphi) \supset \varphi$$

$$P\ \mathsf{reps}\ Q\ \mathsf{on}\ \varphi \stackrel{\text{def}}{=} P \mid Q\ \mathsf{says}\ \varphi \supset Q\ \mathsf{says}\ \varphi$$

Fig. 2. Core Inference Rules

2. Delegates generally should not be able to restrict the scope of their duties as a principal's representative. For example, suppose that Alice delegates to Bob the task of withdrawing $500 from her checking account and depositing it to her savings account. Bob should not be able to withdraw the funds without also depositing them; to do so would be a violation of his responsibilities, not to mention theft.
3. The delegation relationship generally is not transitive: a delegate should not be able to pass on his responsibilities to someone else.

The first property—that recognized delegates should be able to act on behalf of the principals they represent—is reflected by the *Reps* rule stated below and in Figure 3:

$$\textit{Reps}\ \frac{Q\ \mathsf{controls}\ \varphi \quad P\ \mathsf{reps}\ Q\ \mathsf{on}\ \varphi \quad P \mid Q\ \mathsf{says}\ \varphi}{\varphi}$$

This rule states that if Q is authorized to perform φ, P is recognized as Q's delegate on φ, and P requests φ on Q's behalf, then the request for φ should be

$$\textit{Conjunction}\quad \frac{\varphi_1 \qquad \varphi_2}{\varphi_1 \wedge \varphi_2}$$

$$\textit{Simplification (1)}\quad \frac{\varphi_1 \wedge \varphi_2}{\varphi_1} \qquad \textit{Simplification (2)}\quad \frac{\varphi_1 \wedge \varphi_2}{\varphi_2}$$

$$\textit{Quoting (1)}\quad \frac{P \mid Q \text{ says } \varphi}{P \text{ says } Q \text{ says } \varphi} \qquad \textit{Quoting (2)}\quad \frac{P \text{ says } Q \text{ says } \varphi}{P \mid Q \text{ says } \varphi}$$

$$\textit{\&Says (1)}\quad \frac{P\&Q \text{ says } \varphi}{P \text{ says } \varphi \wedge Q \text{ says } \varphi} \qquad \textit{\&Says (2)}\quad \frac{P \text{ says } \varphi \wedge Q \text{ says } \varphi}{P\&Q \text{ says } \varphi}$$

$$\textit{Controls}\quad \frac{P \text{ controls } \varphi \quad P \text{ says } \varphi}{\varphi} \qquad \textit{Derived Speaks For}\quad \frac{P \Rightarrow Q \quad P \text{ says } \varphi}{Q \text{ says } \varphi}$$

$$\textit{Reps}\quad \frac{Q \text{ controls } \varphi \quad P \text{ reps } Q \text{ on } \varphi \quad P \mid Q \text{ says } \varphi}{\varphi}$$

$$\textit{Rep Says}\quad \frac{P \text{ reps } Q \text{ on } \varphi \quad P \mid Q \text{ says } \varphi}{Q \text{ says } \varphi}$$

Fig. 3. Derived Rules Used in this Paper

granted. This rule can be derived from the sound rules of Figure 2 and thus is sound itself.

The second and third properties both state things that should **not** happen. For that reason, it is necessary to verify that our definition of delegation **prohibits** the reduction or passing on of delegation duties. The following two rules, which would allow the undesired behavior, can easily be shown to be unsound with respect to the Kripke semantics:

$$\textit{Unsound Rule!}\quad \frac{P \text{ reps } Q \text{ on } \varphi_1 \wedge \varphi_2}{P \text{ reps } Q \text{ on } \varphi_1}$$

$$\textit{Unsound Rule!}\quad \frac{P \text{ reps } Q \text{ on } \varphi \quad Q \text{ reps } R \text{ on } \varphi}{P \text{ reps } R \text{ on } \varphi}$$

The original papers by Lampson, Abadi and colleagues [3,4] introduced a notion of delegation based on a "fictional delegation server", whose purpose was to co-sign or back up any of a delegate's authentic requests. Their notion of delegation was a universal one: if Bob is Alice's delegate, then Bob represents Alice on *all* statements, not only on specifically designated ones. Furthermore, access policies had to specifically name delegates in addition to the principals they represented (e.g., *Bob* for *Alice* controls φ).

Our definition of delegation is a much simpler one, but we believe that the soundness of the *Reps* rule and the lack of soundness of the two undesired rules

provide supporting evidence of its correctness. The next section demonstrates its suitability for reasoning about fairly complicated situations, such as how checks work as payment instruments within the retail payment system.

3 Checking Using an Electronic Clearing House Network

In this section we describe a banking network that uses electronic credits and debits and the Automated Clearing House (ACH) network—a trusted third-party settlement service. Detailed descriptions of retail payment systems and the ACH network can be found in the Federal Financial Institutions Examination Council's handbook on Retail Payment Systems [7] and the National Automated Clearing House Association's guide to rules and regulations governing the ACH network[5].

We pay particular attention to patterns of description and patterns of reasoning. The patterns of description take the form of definitions—typically formal definitions of financial instruments, statements of jurisdiction, and policy statements. Patterns of reasoning take the form of derived (and thus inherently sound) inference rules that reflect the implicit access-control decisions being made in the retail payment system. We include the formal proofs that justify the access-control decisions being made. These proofs are simple and show explicitly the trust relationships upon which the decisions are being made. We believe the combination of simplicity, precision, clarity, and soundness are of tremendous benefit to systems engineers and certifiers who build and evaluate complicated systems where delegation is widely used.

As we will be focusing on how checks and endorsed checks are used, we give them formal definitions. We adopt the notational convention that atomic (indivisible) actions are surrounded by "⟨" and "⟩". For example, $\langle debit\ \$100, acct_{Alice}\rangle$ is interpreted as the atomic proposition "*it would be a good idea to debit Alice's account by $100.*" To save space, we also adopt the notational abbreviation P controls+says φ to denote the *two* statements: P controls φ and P says φ. We use P controls+says φ in describing policies, but use both P controls φ and P says φ in formal proofs.

Definition 3. *A check is an order from a principal (known as the payer) upon a bank to draw upon the payer's deposit of funds to pay a certain amount of money to another principal (known as the payee). If P is the payer and Q is the payee, we represent a check written by P to Q as follows:*

$$Signature_P \text{ says } (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$$

The check associates P's *signature* with the statement to pay Q and to debit P's account. As our subsequent analysis will show, one must be able to associate statements made by $Signature_P$ with P; this association is represented in the logic as $Signature_P \Rightarrow P$.

Definition 4. *A check is* endorsed *when the payee signs the check issued to him or her. If P is the payer and Q is the payee, we represent a check written by P and endorsed by Q as follows:*

$$Signature_Q \mid Signature_P \text{ says } (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$$

The banking system uses clearing houses (or clearing corporations) to collect and settle individual transactions while minimizing the payments made between banks.

Definition 5. *[7] defines a* clearing corporation *as follows:*

> *A central processing mechanism whereby members agree to net, clear, and settle transactions involving financial instruments. Clearing corporations fulfill one or all of the following functions:*
> - *nets many trades so that the number and the amount of payments that have to be made are minimized,*
> - *determines money obligations among traders, and*
> - *guarantees that trades will go through by legally assuming the risk of payments not made or securities not delivered. This latter function is what is implied when it is stated that the clearing corporation becomes the "counter-party" to all trades entered into its system. Also known as a clearinghouse or clearinghouse association.*

To understand how a clearing house works, suppose that some depositors of $Bank_P$ and $Bank_Q$ exchange a total of two checks as follows during the day:

1. Bob, a $Bank_Q$ depositor deposits a \$100 check from Alice, a $Bank_P$ depositor.
2. Dan, a $Bank_P$ depositor deposits a \$250 check from Carol, a $Bank_Q$ depositor.

$Bank_P$ and $Bank_Q$ send the deposited checks to a clearing house to total up the transactions between them. The clearing house will let each bank know how much it owes (or is owed) to (or from) other banks to settle their accounts each banking day. In this example, $Bank_P$ and $Bank_Q$ settle by having $Bank_Q$ transfer \$150 to $Bank_P$. $Bank_P$ will credit \$250 to Dan's account and debit Alice's account by \$100. $Bank_Q$ will credit \$100 to Bob's account and debit \$250 from Carol's account. In the (hopefully) unlikely event that $Bank_Q$ is unable to cover its debts, the clearing house will pay $Bank_P$ what it is owed.

Another feature that provides the benefits of faster processing of checks to banks and consumers is the practice of *check truncation*.

Definition 6. *[7] defines* check truncation *as follows:*

> *The practice of holding a check at the institution at which it was deposited (or at an intermediary institution) and electronically forwarding the essential information on the check to the institution on which it was written. A truncated check is not returned to the writer.*

Banks and other financial institutions use electronic check conversion (ECC) to convert endorsed physical checks into legally equivalent check images in support of check truncation.

Definition 7. Electronic check conversion *is the process of using magnetic-ink character recognition (MICR) to capture information from a check's MICR line, including: the bank's routing number, account number, check number, check amount, and other information that are printed near the bottom of the check in magnetic ink in accordance with generally applicable industry standards.*

We represent $Bank_Q$'s ECC of an endorsed check as:

$$ECC_{Bank_Q} \text{ says } (Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)$$

When electronically converted checks are used in the context of check truncation, this is known as electronic check presentment.

Definition 8. *[7] defines* electronic check presentment (ECP) *as follows:*

> *Check truncation methodology in which the paper check's MICR line information is captured and stored electronically for presentment. The physical checks may or may not be presented after the electronic files are delivered, depending on the type of ECP service that is used.*

$Bank_Q$'s presentation of the electronic check image can be represented as:

$$Bank_Q \text{ says } (ECC_{Bank_Q} \text{ says } (Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle))$$

The above formula states that $Bank_Q$ is relaying the output of its ECC process. Presumably, $Bank_Q$ only makes this statement (i.e., vouches for its electronic check conversion process) when it believes the process is working correctly.

Figure 4 illustrates the use of a clearing house and check images. Bob, the payee, deposits Alice's check at his bank. Bob's bank does not present the check endorsed by Bob to Alice's bank directly. Rather, Bob's bank *truncates* the check, credits Bob's account, and sends an electronic version of the check (usually in a batch with other orders) to an Automated Clearing House (ACH) operator, who sends the image and information to Alice's bank to debit Alice's account. The ACH operator settles the accounts between Alice's and Bob's respective banks each day.

Clearing corporations such as the Federal Reserve Banks guarantee payments for depository financial institutions using services such as FedACH. Consequently, the Federal Reserve Banks take on the financial risk if a depository financial institution (DFI) defaults and has insufficient funds to settle. Hence, both ACH and the DFIs are signatories to transactions. Thus, ACH is not merely relaying information but assuming liability.

We represent the presentation of a check image created by $Bank_Q \mid ECC_{Bank_Q}$ by an ACH operator ACH as follows:

$$((ACH \& Bank_Q) \mid ECC_{Bank_Q}) \mid Q \text{ says } (Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle))$$

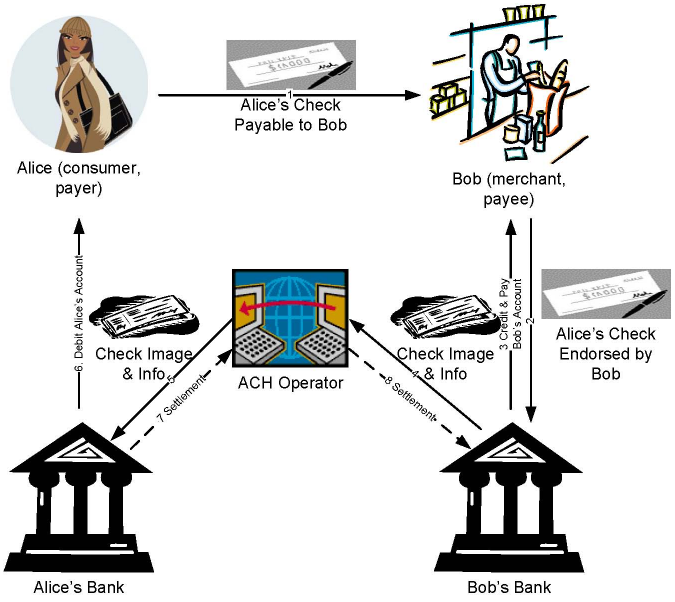

Fig. 4. Interbank Checking Using an Automated Clearing House

The operator is functioning as a clearing corporation and *counter signs* the check image. By so doing, the ACH operator assumes the risk of the transaction if $Bank_P$ defaults at settlement time.

In this system checks are cleared immediately without first checking balances. If there is an insufficient balance to cover the amount of the check, the check in question is returned to the depositor and the amount ultimately charged back to his or her account as a separate transaction. In Figure 4, if Alice's check bounces, then Alice's check (or a truncated version of her check) is returned by her bank to the ACH operator to debit Bob's bank the amount of the returned check.

Authorities, Jurisdiction, and Policies. The controlling authorities in this case include the bank owners with the addition of the Automated Clearing House (ACH) association, whose rules all members agree to follow as a condition of membership. Our analysis starts with $Bank_P$, as it is the paying bank.

$Bank_P$: At the individual account level, depositors are allowed to write checks. If there are insufficient funds in the account, another transaction will reverse the debit. Therefore, the policy allowing checks to be written is given below:

$$Bank_P \ \ Owner \ \mathsf{controls{+}says} \ (P \ \mathsf{controls} \ (\langle Pay \ \text{amt}, Q\rangle \wedge \langle debit \ \text{amt}, acct_P\rangle))$$

The policy allows payees to be delegates of the payers indicated on checks:

$$Bank_P \ \ Owner \ \mathsf{controls{+}says} \ (Q \ \mathsf{reps} \ P \ \mathsf{on} \ (\langle Pay \ \text{amt}, Q\rangle \wedge \langle debit \ \text{amt}, acct_P\rangle))$$

Applying the *Controls* inference rules to the above statements produces the following policy statements for $Bank_P$:

$$P \text{ controls } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)$$
$$Q \text{ reps } P \text{ on } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)$$

Because $Bank_P$ is part of the ACH network, it recognizes ACH as a counter signer with the ACH network banks that use ECP:

$$Bank_P \ \ Owner \text{ controls+says}$$
$$((ACH\&Bank_Q) \mid ECC_{Bank_Q}) \text{ reps } (Q \mid Signature_P) \text{ on}$$
$$(\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)$$

Using the *Controls* inference rule, we derive $Bank_P$'s policy regarding check images and information forwarded to it by ACH:

$$((ACH\&Bank_Q) \mid ECC_{Bank_Q}) \text{ reps } (Q \mid Signature_P) \text{ on}$$
$$(\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)$$

ACH: The ACH operator only accepts transactions from ACH members. In this example, the policies for the ACH operator regarding $Bank_P$ and $Bank_Q$ are as follows:

$$Bank_P \text{ controls } \langle Pay \text{ amt}, Q\rangle$$

The above states that the ACH operator will accept a payment from $Bank_P$ as part of the settlement process. The next formula states that $Bank_Q$ is allowed to present electronically converted checks to the operator:

$$Bank_Q \text{ reps } ECC_{Bank_Q} \text{ on}$$
$$(Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle))$$

$Bank_Q$: The controlling authority for $Bank_Q$ is $Bank_Q$'s owner. The following statements are a result of recognizing $Bank_P$ as a banking partner as part of the ACH network (this would be determined from the MICR line). The first policy states that checks drawn upon accounts in $Bank_P$ may be deposited in $Bank_Q$'s accounts:

$$Bank_Q \ \ Owner \text{ controls+says}$$
$$((Q \mid Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)) \supset$$
$$(\langle Pay \text{ amt}, Q\rangle \wedge (Q \text{ controls } \langle credit \text{ amt}, acct_Q\rangle)))$$

We are assuming here that funds are immediately available (i.e., there is no float time). The second policy states that $Bank_Q$ recognizes ACH's settlement statement:

$$Bank_Q \ \ Owner \text{ controls+says } (ACH \text{ controls } \langle Pay \text{ amt}, Q\rangle)$$

Applying the *Controls* inference rule to the above statements produces the following policies for $Bank_Q$:

$$((Q \mid Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)) \supset (Q \text{ controls } \langle credit \text{ amt}, acct_Q\rangle \wedge \langle Pay \text{ amt}, Q\rangle))$$

and

$$ACH \text{ controls } \langle Pay \text{ amt}, Q\rangle$$

Operating Rules. There are five access-control decisions to be made, corresponding to the arrows labeled 3–8 in Figure 4. The first decision (arrow 3) is made by $Bank_Q$. This decision corresponds to Bob's request to deposit Alice's check, credit his account by the same amount, and have the funds made available to him. This decision is made by the *ACH Check Deposit* rule, whose proof is in Figure 5:

$$\textit{ACH Check Deposit}\quad \frac{\begin{array}{c} Signature_Q \mid Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle) \\ Signature_Q \text{ says } \langle credit \text{ amt}, acct_Q\rangle \\ Signature_Q \Rightarrow Q \\ (Q \mid Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)) \supset \\ \langle Pay \text{ amt}, Q\rangle \wedge (Q \text{ controls } \langle credit \text{ amt}, acct_Q\rangle) \end{array}}{\langle Pay \text{ amt}, Q\rangle \wedge \langle credit \text{ amt}, acct_Q\rangle}$$

The second decision (arrow 4) is also made by $Bank_Q$, which must decide whether or not to electronically present the check endorsed by Q to the ACH operator. If the check is endorsed by a depositor Q of $Bank_Q$, $Signature_Q$ is Q's signature, and the check itself passes whatever integrity check is used by the bank, then the check is converted to its electronic version and passed on to the ACH operator. This decision is made by the *ACH Check Presentation* rule, which is proved in Figure 6:

$$\textit{ACH Check Presentation}\quad \frac{\begin{array}{c} Signature_Q \mid Signature_P \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle) \\ Signature_Q \Rightarrow Q \end{array}}{Bank_Q \text{ says } (ECC_{Bank_Q} \text{ says } (Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle))}$$

The third decision (arrow 5) is made by the ACH operator to counter sign the electronically converted check and present it to $Bank_P$. This decision uses the *ACH Counter Sign* rule, whose proof is in Figure 7:

$$\textit{ACH Counter Sign Rule}\quad \frac{\begin{array}{l} Bank_Q \text{ says } (ECC_{Bank_Q} \text{ says} \\ \quad (Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)) \\ Bank_Q \text{ reps } ECC_{Bank_Q} \text{ on} \\ \quad (Q \mid Signature_P) \text{ says } (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle)) \end{array}}{\begin{array}{l} (ACH \& Bank_Q) \text{ says } (ECC_{Bank_Q} \text{ says } ((Q \mid Signature_P) \text{ says} \\ \quad (\langle Pay \text{ amt}, Q\rangle \wedge \langle debit \text{ amt}, acct_P\rangle))) \end{array}}$$

1. $Signature_Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	Endorsed Check
2. $Signature_Q$ says $\langle credit\ \text{amt}, acct_Q\rangle$	Deposit Slip
3. $Signature_Q \Rightarrow Q$	Signature on File
4. $(Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)) \supset$ $\langle Pay\ \text{amt}, Q\rangle \wedge (Q$ controls $\langle credit\ \text{amt}, acct_Q\rangle)$	$Bank_Q$ Policy
5. $Signature_P \Rightarrow Signature_P$	Idempotency of $\Rightarrow$
6. $Signature_Q \mid Signature_P \Rightarrow Q \mid Signature_P$	Monotonicity of $\mid$
7. $Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	6, 1 Derived Speaks For
8. $\langle Pay\ \text{amt}, Q\rangle \wedge (Q$ controls $\langle credit\ \text{amt}, acct_Q\rangle)$	7, 4 Modus Ponens
9. $\langle Pay\ \text{amt}, Q\rangle$	8 Simplification (1)
10. Q controls $\langle credit\ \text{amt}, acct_Q\rangle$	8 Simplification (2)
11. Q says $\langle credit\ \text{amt}, acct_Q\rangle$	3, 2 Derived Speaks For
12. $\langle credit\ \text{amt}, acct_Q\rangle$	10, 11 Controls
13. $\langle Pay\ \text{amt}, Q\rangle \wedge \langle credit\ \text{amt}, acct_Q\rangle$	9, 12 Conjunction

Fig. 5. Proof of ACH Check Deposit

The fourth decision (arrows 6 and 7) is made by $Bank_P$ to debit the appropriate account and pay toward settlement. This rule (*ACH Check Funding*) is proved in Figure 8:

$$\textit{ACH Check Funding}\quad \frac{\begin{array}{l}(ACH\&Bank_Q)\ \text{says}\ (ECC_{Bank_Q}\ \text{says}\ ((Q \mid Signature_P)\ \text{says}\\ \quad (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)))\\ ((ACH\&Bank_Q) \mid ECC_{Bank_Q})\ \text{reps}\ Q\ \text{on}\\ \quad (Signature_P\ \text{says}\ (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))\\ Signature_P \Rightarrow P\\ P\ \text{controls}\ (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)\\ Q\ \text{reps}\ P\ \text{on}\ (\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)\end{array}}{(Bank_P\ \text{says}\ \langle Pay\ \text{amt}, Q\rangle) \wedge \langle debit\ \text{amt}, acct_P\rangle}$$

The final decision (arrow 8) is made by the ACH operator using the *ACH Check Settlement Rule*, which is proved in Figure 9.

$$\textit{ACH Check Settlement Rule}\quad \frac{\begin{array}{l}Bank_P\ \text{says}\ \langle Pay\ \text{amt}, Q\rangle\\ Bank_P\ \text{controls}\ \langle Pay\ \text{amt}, Q\rangle\end{array}}{ACH\ \text{says}\ \langle Pay\ \text{amt}, Q\rangle}$$

Risks. All of the risks of paper-based checking are present in the ACH system. There is an additional risk incurred here, because in practice the ACH system does not look up signatures to ensure that the signature on a check matches a signature on file. Instead, checking a signature occurs only when a customer complains about a fraudulent check.

Despite the potential risks, the ACH system is largely trustworthy. The ACH system is highly automated and runs on exceptions, so that large numbers of transactions are cleared daily. Transactions are largely assumed to be legitimate, and the evidence in terms of the number of checks returned unpaid supports this. The 2004 Federal Reserve Payments Study [8] reported:

> *In 2000, the number of checks returned unpaid was 0.6 percent of checks paid by depository institutions, compared to 0.5 percent in 2003. The value per returned check has remained relatively unchanged: $756 [in 2003] compared to $747 . . . [in 2000].*

1. $Signature_Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	Endorsed Check
2. $Signature_Q \Rightarrow Q$	Signature on File
3. $Signature_P \Rightarrow Signature_P$	Idempotency of $\Rightarrow$
4. $Signature_Q \mid Signature_P \Rightarrow Q \mid Signature_P$	Monotonicity of $\mid$
5. $Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	4, 1 Derived Speaks For
6. ECC_{Bank_Q} says $(Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	6 Says
7. $Bank_Q$ says ECC_{Bank_Q} says $(Q \mid Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	7 Says

Fig. 6. Proof of ACH Check Presentation

1. $Bank_Q$ says $(ECC_{Bank_Q}$ says $(Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	Presented Check
2. $Bank_Q$ reps ECC_{Bank_Q} on $(Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	ACH policy
3. $Bank_Q \mid ECC_{Bank_Q}$ says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	1 Quoting (2)
4. ECC_{Bank_Q} says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	2, 3 Rep Says
5. ACH says $(ECC_{Bank_Q}$ says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)))$	4 Says
6. $ACH\&Bank_Q$ says $(ECC_{Bank_Q}$ says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)))$	5, 1 &Says (2)

Fig. 7. Proof of ACH Counter Sign Rule

1. $(ACH\&Bank_Q)$ says $(ECC_{Bank_Q}$ says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)))$	Check presented by ACH
2. $((ACH\&Bank_Q) \mid ECC_{Bank_Q})$ reps Q on $(Signature_P$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	$Bank_P$ policy
3. $Signature_P \Rightarrow P$	Signature on File
4. P controls $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	$Bank_P$ policy
5. Q reps P on $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle)$	$Bank_P$ policy
6. $(ACH\&Bank_Q \mid ECC_{Bank_Q})$ says $((Q \mid Signature_P)$ says $(\langle Pay\ \text{amt}, Q\rangle \wedge \langle debit\ \text{amt}, acct_P\rangle))$	1 Quoting (2)

Fig. 8. Proof of ACH Check Funding

1. $Bank_P$ says $\langle Pay\ \text{amt}, Q\rangle$	$Bank_P$ authorizing payment to Q
2. $Bank_P$ controls $\langle Pay\ \text{amt}, Q\rangle$	ACH's policy to rely/trust in $Bank_P$'s authorization
3. $\langle Pay\ \text{amt}, Q\rangle$	2, 1 Controls
4. ACH says $\langle Pay\ \text{amt}, Q\rangle$	3 says

Fig. 9. Proof of ACH Check Settlement Rule

The ACH rules [5] and Check 21 regulations [6] add consistency and uniformity of operations and formats that supports delegation and trust. Speeding up the processing of checks due to check truncation reduces the float time for checks and works against frauds such as check kiting. Guarding against fraud ultimately depends on the consumer's awareness and diligence in monitoring transactions. The ability, willingness, and speed with which consumers are able to detect "exceptions" varies and is likely the largest risk in the payment system.

4 Conclusions

Systems engineering is difficult in part because of the myriad interacting components and the often crucial and undocumented assumptions about the context in which a component, subsystem or system will operate. (A particularly stunning example was the loss of the Mars Climate Orbiter due to Lockheed Martin engineers assuming an English interpretation of data while NASA engineers assumed a metric interpretation). In systems where delegates are used extensively and access-control decisions are made automatically, it is crucial to understand and precisely document how and why these decisions are made. Understanding of complicated systems is best served when the underlying formal system of reasoning is simple yet effective. We believe our logic for delegation and trust has these properties and is a useful tool for systems engineers and system certifiers.

Regarding the check-based retail payment system itself, using the access-control logic brings the intricacies of trust and delegation to the fore. These intricacies include precise statements about who is trusted on precisely what statements. Using the logic and delegation definitions, we are able to formally justify what amounts to the access-control decisions made in the retail payment system. Not only does this provide insight into how the system works, it also provides a precise specification for how the check-based system should behave. While we have focused entirely on paper and electronic checks, a similar description and analysis should hold for other payment instruments, such as credit cards, debit cards, and stored-value cards.

Our long-term goal is to provide to systems engineers a similar combination of accessibility, usability, and rigor that hardware engineers enjoy with digital logic. In particular, digital logic is the foundation of hardware design and verification. It is introduced at the introductory undergraduate level to explain core concepts with precision. At more advanced levels, digital logic provides the formal basis for computer-aided design tools such as verifiers and synthesizers.

We have used our access-control logic to describe a variety of systems, including everyday situations such as airport security, web-based services such as CORBA [1], control of physical memory [9,10], and role-based access control [2]. We have incorporated our logic into courses at both the undergraduate and graduate levels [11,10]. In our experience, this logic is accessible to rising juniors and seniors in computer science and computer engineering. These experiences lead us to believe that the goal of giving engineers simple, effective, and mathematically sound tools for assuring security is feasible and within reach.

References

1. Kosiyatrakul, T., Older, S., Humenn, P.R., Chin, S.K.: Implementing a calculus for distributed access control in higher order logic and hol. In: Gorodetsky, V., Popyack, L.J., Skormin, V.A. (eds.) MMM-ACNS 2003. LNCS, vol. 2776, pp. 32–46. Springer, Heidelberg (2003)
2. Kosiyatrakul, T., Older, S., Chin, S.K.: A modal logic for role-based access control. In: Gorodetsky, V., Kotenko, I.V., Skormin, V.A. (eds.) MMM-ACNS 2005. LNCS, vol. 3685, pp. 179–193. Springer, Heidelberg (2005)

3. Lampson, B., Abadi, M., Burrows, M., Wobber, E.: Authentication in Distributed Systems: Theory and Practice. ACM Transactions on Computer Systems 10(4), 265–310 (1992)
4. Abadi, M., Burrows, M., Lampson, B., Plotkin, G.: A Calculus for Access Control in Distributed Systems. ACM Transactions on Programming Languages and Systems 15(4), 706–734 (1993)
5. National Automated Clearing House Association 13665 Dulles Technology Drive, Suite 300, Herndon, VA 20171: 2006 ACH Rules: A Complete Guide to Rules and Regulations Governing the ACH Network (2006)
6. 108th Congress (Check 21 act) Public Law Number 108–100, 117 Stat (2003). Public Law 108–100 was the 100th law passed by the 108th Congress. It was published in vol. 117, p. 1177 (2003) of the United States Statutes at Large at available at `http://www.federalreserve.gov/paymentsystems/truncation/`
7. Federal Financial Institutions Examination Council: Retail Payment Systems: IT Examination Handbook (2004) Available under IT Booklets on the FFIEC IT Handbook InfoBase web page at `http://www.ffiec.gov/`
8. Federal Reserve System: The 2004 Federal Reserve Payments Study: Analysis of Noncash Payments Trends in the United States: 2000–2003 (2004) Available at `www.frbservices.org/Retail/pdf/2004PaymentResearchReport.pdf`
9. Saltzer, J., Schroeder, M.: The protection of information in computer systems. In: Proceedings of IEEE 1975, IEEE Computer Society Press, Los Alamitos (1975)
10. Chin, S.K., Older, S.: A rigorous approach to teaching access control. In: Proceedings of the First Annual Conference on Education in Information Security, ACM, New York (2006)
11. Older, S., Chin, S.K.: Using Outcomes-based Assessment as an Assurance Tool for Assurance Education. Journal of Information Warfare 2(3), 86–100 (2003)