Using Trust Management to Support Transferable Hash-Based Micropayments

Simon N. Foley

Department of Computer Science,
University College, Cork, Ireland
s.foley@cs.ucc.ie

Abstract. A hash-chain based micropayment scheme is cast within a trust management framework. Cryptographic delegation credentials are used to manage the transfer of micropayment contracts between public keys. Micropayments can be efficiently generated and determining whether a contract and/or micropayment should be trusted (accepted) can be described in terms of a trust management compliance check. A consequence is that it becomes possible to consider authorisation based, in part, on monetary concerns. The KeyNote trust management system is used to illustrate the approach.

Keywords: Delegation; Digital Cash; One-way Hash Functions; Public Key Certificates; Trust Management.

1 Introduction

Trust Management [3,4,9,21] is an approach to constructing and interpreting the trust relationships between public keys that are used to mediate security critical actions. Cryptographic credentials are used to specify delegation of authorisation among public keys.

In this paper we consider how Trust Management can be used to manage trust relationships that are based on monetary payment. A benefit to characterising a payment scheme as a trust management problem is the potential to support, within the trust management framework, sophisticated trust requirements that can combine monetary and authorisation concerns. For example, authorisation to access a valuable resource might be based on some suitable combination of permission and monetary payment or deposit (possibly partially refundable if it can be determined that the resource is not misused). In this case, perhaps a user with ‘less’ permissions would be required to provide a larger deposit, while a user with ‘more’ permissions provides a smaller deposit. The trust management system is expected to help the application system manage what is meant by trusted access to this resource.

In [7,14], the KeyNote trust management system is used to manage trust for a micro-billing based payment scheme. Their scheme is similar to IBM’s mini-pay scheme [13]: KeyNote is used by the payer (merchant) to determine whether or not an off-line payment from a particular payee (customer) should...
be trusted, or whether the payee should go online to validate the payment and
payee. The scheme is intended for small value payments (under $1.00). Since
the generation of each payment transaction requires a public key cryptographic
operation (signature) then it may not be practical for very low value payments
where the cost of processing is high relative to the value of the payment.

This paper extends earlier work [11] exploring how KeyNote can be used
to support low-value micropayments in the provision of authorised access
to resources by the participants of a meta-computer The meta-computer [10, 
[18] is a network of heterogenous computers that work together to solve large
problems. It encourages the sharing of resources between different organisations,
departments and individuals. A consequence of this sharing is that the providers
of the resources expect payment for their computation processing and services,
on a per-use basis. For example, participants might be paid for contributing
processing power to weather model computation.

In [11], we describe a preliminary KeyNote implementation of micropayments
that is based on hash-chains [18,20]. These schemes are limited in that it is not
possible for the participants to transfer (delegate) their micropayment contracts
to third parties. Some form of efficient transfer is desirable. For example, an
overloaded client workstation might like to transfer some of its workload and
the corresponding micropayment contract that it holds to another client.

The contribution in this paper is the development of a hash-chain based
micropayment scheme that is cast within a trust management framework and
supporting the efficient transfer of micropayment contracts. Codifying a mi-
cropayment scheme in terms of KeyNote also demonstrates the usefulness and
applicability of trust management systems in general.

The paper is organised as follows. Section 2 describes a simple model of
cryptographic credential based delegation that forms the basis of trust manage-
ment. Section 3 extends this model by considering how values (that represent
permissions) along a hash-chain can be delegated without having to sign new
credentials each time. This extension forms the basis of the transferable mi-
cropayment scheme that is proposed in Section 4. Section 5 considers how the
scheme can be codified and interpreted within the KeyNote trust management
system.

2 Delegating Authorisation

A simple model is used to represent delegation of authorisation between public
dkeys. A signed cryptographic credential, represented as \( \langle K_B, p \rangle_{sK_A} \), indicates
that \( K_A \) delegates to \( K_B \), the authorisation permission \( p \). Permissions are struc-
tured in terms of lattice \( (PERM, \leq, \sqcap) \), whereby \( p \leq q \) means that permission \( q \)
provides no less authorisation than \( p \). A simple example is the power-set lattice
of \{read, write\}, with ordering defined by subset, and greatest lower bound \( (\sqcap) \)
defined by intersection.

Given a credential \( \langle K_B, q \rangle_{sK_A} \) and \( p \leq q \) then there is an implicit delegation
of \( p \) to \( K_B \) by \( K_A \), written as \( \langle K_B, p \rangle_{K_A} \). Two reduction rules follow.