A Model for Distribution and Revocation of Certificates

Åsa Hagström[1] and Francesco Parisi-Presicce[2]

[1] Lawson Software
Åsa.Hagström@se.lawson.com
[2] Dipartimento di Informatica, Sapienza Universitá di Roma
parisi@di.uniroma1.it

Abstract. The distribution and revocation of public-key certificates are essential aspects of secure digital communication. As a first step towards a methodology for the development of reliable models, we present a formalism for the specification and reasoning about the distribution and revocation of public keys, based on graphs. The model is distributed in nature; each entity can issue certificates for public keys that it knows, and distribute these to other entities. Each entity has its own public key bases and can derive new certificates from this knowledge. If some of the support for the derived knowledge is revoked, then some of the derived certificates may be revoked as well. Cyclic support is avoided. Graph transformation rules are used for the management of the certificates, and we prove soundness and completeness for our model.

1 Introduction

A certificate is an assertion, made by an entity (often an "authority"), about some characteristics or privileges of another entity. We are only interested in "public-key certificates" where an asserrtion is the binding of a particular public-key with an entity, assertion signed by another entity with her private-key. In this paper we present a conceptual model to better understand the semantics of certificate distribution and revocation. The model is a graph which describes the knowledge of all entities in a system, and is distributed in nature. We **do not** assume a hierarchical model with a central Certification Authority (CA), but allow every entity to issue and distribute its own certificates to others.

The Merriam-Webster dictionary explains the act of revoking as "to annul by recalling or taking back". Thus, revocation of a certificate could be the act of a user who recalls a certificate previously passed to another user. Somehow, the revocation must cascade in the system to make sure that no information is derived from obsolete certificates.

This description of revocation seems simple enough. However, even in such a specific environment as a public-key infrastructure (PKI) – where all the information consists of certificates, each on the same form – one has to be very careful when defining *what* is being revoked. Similarly, there can be a number of reasons for a revocation. We show how different reasons can be modelled as specific actions, corresponding to the annulment of a specific piece of information.

H. Ehrig et al. (Eds.): ICGT 2010, LNCS 6372, pp. 314–329, 2010.

A stronger way to revoke a certificate is to issue its *inverse*; if there was previously a certificate (signed by A with her private key A.pr) binding B and his public key B.pu, the inverse is a certificate stating that B.pu is *not* B's public key. This annulment is time-persistent in the sense that any subsequent positive certificates for the same binding has to deal with the presence of the negative certificate. We consider any kind of annulment of information – whether by removal or by issuing the inverse – to be a form of revocation.

Our aim is to understand the meaning of revocation in the context of a PKI. The purpose is not to find an efficient implementation for revocation (in particular, we do not deal with Certificate Revocation Lists (CRL's)), but to investigate the implications and how they depend on the reason for the revocation. We also have ideas for including trust statements, but space limits us to key certificates. Many researchers use graphs to exemplify and concretize their ideas. We believe that graphs themselves constitute a powerful tool for modelling and reasoning about systems, and we have chosen to take advantage of their expressive and intuitive properties. Our formalism of choice is a graph which captures the information state of a system, and graph transformation rules which define allowed changes to the information, as well as deductions adding new knowledge.

2 Related Work

The notion of revocation can be given various interpretations. The desired result of revoking a certificate will typically depend on the reason for it. Cooper [1] divides revocation reasons into benign and malicious types, and notes that different revocation practices are needed for the two types. Both Fox and LaMacchia [2] and Gunter and Jim [3] discuss different reasons for revocation, and suitable mechanisms for each case. As for X.509, Housley et al. [4] define nine reason codes for revocation of a public-key certificate (keyCompromise, cACompromise, affiliationChanged, superseded, cessationOfOperation, certificateHold, removeFromCRL, privilegeWithdrawn, aACompromise), but do not suggest different revocation practices for the different codes. In [5], Hagström et al. define and classify eight different types of revocation schemes for an ownership-based access control system using the dimensions resilience, propagation and dominance. Some ideas from all this work will be applied in our model.

Other researchers use graphs in some form or other to visualize their ideas. When it comes to the formal treatment, however, most previous work in this area has used logic-, calculus- or language-based approaches:

Logic-based formalisms
Maurer [6] was one of the first to model a PKI using both keys and trust. *A* both needs to know *B*'s public key and trust him to believe the statements he makes. Every statement is about keys or trust. Trust is given in levels; a higher level of trust in a user implies the possibility of longer chains of derived statements starting in that user. Each user's view (including all belief and trust the user has, and all recommendations made to him) is modelled separately from the