Fine-Grained Control of Security Capabilities

DAN BONEH
Stanford University
XUHUA DING
Singapore Management University
and
GENE TSUDIK
University of California, Irvine

We present a new approach for fine-grained control over users' security privileges (fast revocation of credentials) centered around the concept of an on-line semi-trusted mediator (SEM). The use of a SEM in conjunction with a simple threshold variant of the RSA cryptosystem (mediated RSA) offers a number of practical advantages over current revocation techniques. The benefits include simplified validation of digital signatures, efficient certificate revocation for legacy systems and fast revocation of signature and decryption capabilities. This paper discusses both the architecture and the implementation of our approach as well as its performance and compatibility with the existing infrastructure. Experimental results demonstrate its practical aspects.

Categories and Subject Descriptors: E.3 [**Data Encryption**]—*Public key cryptosystems*; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

General Terms: Algorithms, Security

Additional Key Words and Phrases: Certificate Revocation, Digital Signatures, Public Key Infrastructure

1. INTRODUCTION

We begin this article with an example to illustrate the premise of this work. Consider an organization—industrial, government, or military—where all employees (referred to as *users*) have certain authorizations. We assume that a Public Key Infrastructure (PKI) is available and all users have digital signature,

This work was supported by the Defense Advanced Project Agency (DARPA) under contract F30602-99-1-0530. An earlier version of this paper was presented, in part, at the 2001 Usenix Security Symposium.
Authors' addresses: D. Boneh, Stanford University, Computer Science Dept., Gates 475, Stanford, CA 94305-9045; email: dabo@cs.stanford.edu; X. Ding, School of Information Systems, Singapore Management University, Singapore 25976; email: xuhuad@smu.edu.sg; G. Tsudik, School of ICS, 458 CS Building, Irvine CA 92697-3425; email: gts@ics.usi.edu.

as well as en/de-cryption, capabilities. In the course of performing routine everyday tasks, users take advantage of secure applications, such as email, file transfer, remote log-in and web browsing.

Now suppose that a trusted user (Alice) does something that warrants immediate revocation of her security privileges. For example, Alice might be fired, or she may suspect that her private key has been compromised. Ideally, immediately following revocation, the key holder, either Alice herself or an attacker, should be unable to perform any security operations or use any secure applications. Specifically, this might mean:

—The key holder cannot read any secure email. This includes encrypted email that already resides on Alice's email server (or local host) and possible future email erroneously encrypted for Alice. Although encrypted email may be delivered to Alice's email server, the key holder should be unable to decrypt it.
—The key holder cannot generate valid digital signatures on any further messages. However, signatures generated by Alice prior to revocation may need to remain valid.
—The key holder cannot authenticate itself to corporate servers (and other users) as a legitimate user.

Throughout the paper, we use email as an example application. While it is a popular mechanism for general-purpose communication, our rationale also applies to other secure means of information exchange.

To provide immediate revocation it is natural to first consider traditional revocation techniques. Many revocation methods have been proposed; they can be roughly classified into two prominent types: 1) explicit revocation structures such as *Certificate Revocation Lists (CRLs)* and variations on the theme, and 2) real time revocation checking such as the *Online Certificate Status Protocol (OCSP)* [Myers et al. 1999] and its variants. In both cases, some trusted entities are ultimately in charge of validating user certificates. However, the above requirements for immediate revocation are impossible to satisfy with existing techniques. This is primarily because they do not provide fine-grained enough control over users' security capabilities. Supporting immediate revocation with existing revocation techniques would result in heavy performance cost and very poor scalability, as discussed in Section 8.

As pointed out in McDaniel and Rubin [2000], since each revocation technique exhibits a unique set of pros and cons, the criteria for choosing the best technique should be based on the specifics of the target application environment. Fast revocation and fine-grained control over users' security capabilities are the motivating factors for our work. However, the need for these features is clearly not universal since many computing environments (e.g., a typical university campus) are relatively "relaxed" and do not warrant employing fast revocation techniques. However, there are plenty of government, corporate and military settings where fast revocation and fine-grained control are very important.

Organization. This paper is organized as follows. The next section provides an overview of our work. The technical details of the architecture are presented

in Section 3 and Section 4, respectively. Then, Section 5 shows four extensions. Sections 6 and 7 describe the implementation and performance results, respectively. A comparison with current revocation techniques is presented Section 8, followed by the overview of related work in Section 8.2 and a summary in Section 9.

2. OVERVIEW

We refer to our approach as the SEM architecture. The basic idea is as follows: We introduce a new entity, referred to as a SEM (SEcurity Mediator): an online semi-trusted server. To sign or decrypt a message, a client must first obtain a message-specific token from its SEM. Without this token, the user cannot accomplish the intended task. To revoke the user's ability to sign or decrypt, the security administrator instructs the SEM to stop issuing tokens for that user's future request. At that instant, the user's signature and/or decryption capabilities are revoked. For scalability reasons, a single SEM serves many users.

We stress that the SEM architecture is transparent to non-SEM users—a SEM is not involved in encryption or signature verification operations. With SEM's help, a SEM client (Alice) can generate standard RSA signatures, and decrypt standard ciphertext messages encrypted with her RSA public key. Without SEM's help, she cannot perform either of these operations. This backwards compatibility is one of our main design principles.

Another notable feature is that a SEM is not a *fully* trusted entity. It keeps no client secrets and all SEM computations are checkable by its clients. However, a SEM is *partially trusted* since each signature verifier implicitly trusts it to have checked the signer's (SEM's client's) certificate status at signature generation time. Similarly, each encryptor trusts a SEM to check the decryptor's (SEM's client's) certificate status at message decryption time. We consider this level of trust reasonable, especially since an SEM serves a multitude of clients and thus represents an organization (or a group).

In order to experiment and gain practical experience, we prototyped the SEM architecture using the popular OpenSSL library. SEM is implemented as a daemon process running on a secure server. On the client side, we built plug-ins for the Eudora and Outlook email clients for signing outgoing, and decrypting incoming, emails. Both of these tasks are performed with the SEM's help. Consequently, signing and decryption capabilities can be easily revoked.

It is natural to ask whether the same functionality can be obtained with more traditional security approaches to fine-grained control and fast credential revocation, such as Kerberos. Kerberos [Neuman and Ts'o 1994], after all, has been in existence since the mid-80s and tends to work very well in corporate-style settings. However, Kerberos is awkward in heterogeneous networks such as the Internet; its inter-realm extensions are difficult to use and require a certain amount of manual setup. Furthermore, Kerberos does not inter-operate with modern PKIs and does not provide the universal origin authentication offered by public key signatures. On the other hand, the SEM architecture is fully compatible with existing PKI systems. In addition, the SEM is only