Information Processing Letters 126 (2017) 7–11

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Issuer-free oblivious transfer with access control revisited

Alfredo Rial

University of Luxembourg, Luxembourg

ARTICLE INFO

Article history:
Received 16 March 2016
Received in revised form 21 May 2017
Accepted 22 May 2017
Available online 26 May 2017
Communicated by L. Viganò

Keywords:
Cryptography
Oblivious transfer
Access control
Ideal functionality

ABSTRACT

Oblivious transfer with access control (OTAC) is an extension of oblivious transfer where each message is associated with an access control policy. A receiver can obtain a message only if her attributes satisfy the access control policy for that message. In most schemes, the receiver's attributes are certified by an issuer. Recently, two Issuer-Free OTAC protocols have been proposed. We show that the security definition for Issuer-Free OTAC fulfilled by those schemes poses a problem. Namely, the sender is not able to attest whether a receiver possesses a claimed attribute. Because of this problem, in both Issuer-Free OTAC protocols, any malicious receiver can obtain any message from the sender, regardless of the access control policy associated with the message. To address this problem, we propose a new security definition for Issuer-Free OTAC. Our definition requires the receiver to prove in zero-knowledge to the sender that her attributes fulfill some predicates. Our definition is suitable for settings with multiple issuers because it allows the design of OTAC protocols where the receiver, when accessing a record, can hide the identity of the issuer that certified her attributes.

1. Introduction

Oblivious transfer (OT) [1] is a two-party protocol between a sender and a receiver. The sender receives as input N messages $(m_1, \ldots, m_N)$, while the receiver gets K selection values $(\sigma_1, \ldots, \sigma_K)$. As output, the receiver gets the messages $(m_{\sigma_1}, \ldots, m_{\sigma_K})$. Sender security requires that the receiver gets no information on the other messages, while receiver privacy requires that the sender does not learn any information on $(\sigma_1, \ldots, \sigma_K)$.

Oblivious transfer with access control (OTAC) [2] allows the sender to control access to the messages. The sender receives as input $(m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$, where $(\mathbb{P}_1, \ldots, \mathbb{P}_N)$ are access control policies for each of the messages. Each receiver possesses a set of attributes $\mathbb{A}$ and is able to obtain the message m_i only if $\mathbb{A}$ satisfies $\mathbb{P}_i$.

OTAC schemes involve three types of parties: the sender, who possesses a database $(m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$; the issuer, who certifies the receivers' attributes $\mathbb{A}$ and issues credentials to receivers; the receivers, who first get their attributes certified by the issuer and subsequently employ the issued credentials to access the sender's database.

Receiver privacy requires that the sender does not learn any information on the messages the receiver obtains or on the receiver's attributes. Sender privacy requires that the receiver does not learn any information on messages that were not requested or on messages whose access control policy is not fulfilled by the receiver's attributes. Additionally, in some schemes, the access control policies are hidden from the receivers [3], while in other schemes they are public [2,4]. We describe in detail the security definition for OTAC with public policies in Section 2.

Recently, Guleria and Dutta propose Issuer-Free OTAC with public policies [5,6]. In Issuer-Free OTAC, the role of the issuer is performed by the sender. In this paper, we show that the security definition for issuer-free OTAC in [5,6] poses a problem. In a nutshell, the security definition for OTAC with public policies proposed by Camenisch et al. [2] allows the issuer to learn the receiver's identity and

E-mail address: alfredo.rial@uni.lu.

http://dx.doi.org/10.1016/j.ipl.2017.05.006
0020-0190/

the receiver's attributes in order to attest whether the receiver indeed possesses those attributes. In contrast, in the security definition in [5,6], to protect receiver privacy, the sender learns neither the receiver's identity nor the receiver's attributes, and thus is not able to attest whether the receiver possesses the claimed attributes.

This has serious implications on the security of the protocols proposed in [5,6]. In those protocols, the sender *always* proceeds as if the receiver did possess those attributes without performing any check. This allows any malicious receiver to be issued any attribute, which allows this receiver to obtain any message from the sender, regardless of the access control policy associated with the message. Consequently, the protocols in [5,6] do not enforce any form of access control.

We propose a new security definition for Issuer-Free OTAC. Our definition allows the receiver to prove in zero-knowledge to the sender that her attributes fulfill some predicates. The concrete predicates will depend on the information the sender needs in order to attest the receiver's attributes. In the typical setting where attributes need to be certified by an issuer, we show that our new functionality is useful to handle multiple issuers.

2. Oblivious transfer with access control

Camenisch et al. [2] propose an ideal functionality $\mathcal{F}_{\text{OTAC}}$ for oblivious transfer with access control (OTAC). In this section, we recall that ideal functionality. The interaction between $\mathcal{F}_{\text{OTAC}}$, the sender $\mathcal{E}$, the issuer $\mathcal{I}$, and the receivers $\mathcal{R}_1, \ldots, \mathcal{R}_M$ takes place through the interfaces initdb, issue and transfer. The sender $\mathcal{E}$ possesses a list of messages $(m_1, \ldots, m_N)$. These messages are associated with the access control policies $(\mathbb{P}_1, \ldots, \mathbb{P}_N)$. An access control policy describes the attributes that a receiver must possess in order to be allowed to obtain a message. The attributes that a receiver possesses are certified by $\mathcal{I}$. $\mathcal{F}_{\text{OTAC}}$ maintains an initially empty set $\mathbb{A}_m (m \in [1, M])$ for each of the receivers $\mathcal{R}_m$.

Functionality $\mathcal{F}_{\text{OTAC}}$

1. On input $(\text{initdb}, m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$ from $\mathcal{E}$, $\mathcal{F}_{\text{OTAC}}$ stores $(m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$.
2. On input (issue, a) from $\mathcal{R}_m$, $\mathcal{F}_{\text{OTAC}}$ sends $(\text{issue}, \mathcal{R}_m, a)$ to $\mathcal{I}$. $\mathcal{I}$ sends back a bit b. If $b = 1$, $\mathcal{F}_{\text{OTAC}}$ adds the attribute a to $\mathbb{A}_m$ and sends b to $\mathcal{R}_m$, else $\mathcal{F}_{\text{OTAC}}$ simply sends b to $\mathcal{R}_m$.
3. On input $(\text{transfer}, \sigma)$ from $\mathcal{R}_m$, $\mathcal{F}_{\text{OTAC}}$ proceeds as follows. If $(m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$ is stored, $\mathcal{F}_{\text{OTAC}}$ sends transfer to $\mathcal{E}$. $\mathcal{E}$ sends back a bit b. If $b = 1$ and the attribute set $\mathbb{A}_m$ fulfills the policy $\mathbb{P}_\sigma$, $\mathcal{F}_{\text{OTAC}}$ sends the message m_σ to $\mathcal{R}_m$. If $b = 0$ or if $(m_1, \mathbb{P}_1, \ldots, m_N, \mathbb{P}_N)$ is not stored, $\mathcal{F}_{\text{OTAC}}$ sends $\perp$ to $\mathcal{R}_m$.

As described in [2], $\mathcal{F}_{\text{OTAC}}$ guarantees the following security properties:

Receiver Privacy. When a receiver $\mathcal{R}_m$ obtains a message m_σ, the sender $\mathcal{E}$ learns neither $\mathcal{R}_m$ nor σ, i.e., in the transfer phase, the receiver remains anonymous and the sender does not learn the message that the receiver obtains. The sender only learns that an unknown receiver gets a message whose access control policy is fulfilled by the receiver's attributes.

Sender Security. A corrupt receiver cannot obtain a message whose access control policy is not fulfilled by the receiver's attributes. Colluding receivers are not able to share their attributes, i.e., a group of colluding receivers is not able to get access to a message whose access control policy is not fulfilled by the attributes of a particular receiver in the group. If a corrupt receiver colludes with the issuer, then the receiver can obtain one record at each transfer phase.

3. Issuer-free oblivious transfer with access control in [5, 6]

We recall the ideal functionality $\mathcal{F}_{\text{IOTAC}}$ for issuer-free OTAC proposed by Guleria and Dutta [5,6]. The difference between $\mathcal{F}_{\text{IOTAC}}$ and the functionality $\mathcal{F}_{\text{OTAC}}$ described in Section 2 is in the issuing phase. Therefore, we only recall the issue interface of $\mathcal{F}_{\text{IOTAC}}$.

Functionality $\mathcal{F}_{\text{IOTAC}}$: interface issue

2. On input (issue, a) from $\mathcal{R}_m$, $\mathcal{F}_{\text{IOTAC}}$ sends issue to $\mathcal{E}$. $\mathcal{E}$ sends back a bit b in response to issue. If $b = 1$, $\mathcal{F}_{\text{IOTAC}}$ adds the attribute a to $\mathbb{A}_m$ and sends b to $\mathcal{R}_m$, else $\mathcal{F}_{\text{IOTAC}}$ does nothing.

As can be seen, in $\mathcal{F}_{\text{IOTAC}}$, in contrast to $\mathcal{F}_{\text{OTAC}}$, the issuer is not present and the issuing phase is executed by the sender $\mathcal{E}$ and by the receiver $\mathcal{R}_m$. Additionally, while in $\mathcal{F}_{\text{OTAC}}$ the issuer receives the identity of the receiver $\mathcal{R}_m$ and the attribute a, in $\mathcal{F}_{\text{IOTAC}}$ the sender receives neither $\mathcal{R}_m$ nor a.

The latter difference creates a problem. In a real protocol that realizes $\mathcal{F}_{\text{OTAC}}$, the issuer can receive the identity of the receiver $\mathcal{R}_m$ and the attribute a. Based on that information, the issuer is able to attest whether $\mathcal{R}_m$ possesses the attribute a, and, in that case, the issuer issues a credential on that attribute to $\mathcal{R}_m$. However, in any real protocol that realizes $\mathcal{F}_{\text{IOTAC}}$, the sender cannot receive any information on $\mathcal{R}_m$ or a whatsoever. (The reason is that, in the ideal protocol, the sender does not receive that information.) In that case, how is the sender supposed to decide whether the receiver possesses that attribute? This has serious implications on the security of the real world protocols that realize $\mathcal{F}_{\text{IOTAC}}$ proposed in [5,6]. In those protocols, the sender does not receive any information on the attributes or on the identity of the receiver in the issuing phase, and in fact the sender *always* proceeds as if the receiver did possess those attributes without performing any check. This allows any malicious receiver to be