Anonymous Authentication Scheme with Decentralized Multi-authorities

Hiroaki Anada
Department of Information Security
University of Nagasaki
Nagasaki, Japan
Email: anada@sun.ac.jp

Seiko Arita
Graduate School of Information Security
Institute of Information Security
Yokohama, Japan
Email: arita@iisec.ac.jp

***Abstract*—We propose an anonymous authentication scheme with a feature that more than one authorities such as license issuers or product providers can admit a single entity by issuing secret keys, and then the entity is able to prove that is has those secret keys associated to its identity, without a central authority. First, we provide the syntax and the security definition of an anonymous authentication scheme with such decentralized multi-authorities. Next, we give a construction of an anonymous authentication scheme in the discrete logarithm setting by using the Okamoto identification scheme and the Pedersen commitment scheme as building blocks. Then we prove that, under the discrete logarithm assumption, our scheme possesses the proving ability of knowing secret keys associated to the single identity, the anonymity, and the security against concurrent attacks of causing misauthentication. The algorithm of our scheme does not need costly pairing computation, and hence our scheme is suitable for devices with less computational resource.**

I. Introduction

Authentication is a process of determining whether someone or something is in fact the one which has been beforehand admitted by an authority. Currently, logging into a service on a website needs inputting a pair of ID and a password, and once the pair matches with a string on a database the process successfully ends. The succeeded entity goes into the next process of getting permission to do something on the service.

The recent growth of the internet of things and related big data analysis makes protecting privacy more important, and hence the framework of ID and a password should be replaced by an anonymous authentication mechanism. Another direction is a multi-factor authentication scheme for more security to prevent misauthentication. In the scheme an entity is granted access only after successfully presenting several separate pieces of evidence. Actually the multi-factor authentication of using both a laptop PC and a smartphone is getting popular.

In this paper, we propose an anonymous authentication scheme with a feature that various authorities such as license issuers or product providers can admit a person or a thing without a central authority. For example, a person identified by a string like a social security number or a passport ID number is authorized by several organizations of tax, hospital, job opportunity, immigration control, etc. Those organizations are independent and issue authorization evidence separately. Here the need of authorization by decentralized multi-authorities arises. Note here that we must care the privacy problem because if the kind of number like the social security number is once revealed, then his records of activities are linked and analysed for serious purpose like job interview or unexpected sales marketing. In those situation, our anonymous authentication scheme with decentralized multi-authorities is useful.

A. Related Works, our Contribution and Construction

Anonymous credential systems (AC for short) share the same spirit with our proposal. Sadiah et al. [15] proposed AC for proving possession of credentials that satisfy a monotone formula given to both a prover and a verifier. Camenisch et al. [7] proposed AC for multi-authorities with the universal composability. Attribute-based identification schemes (ABID for short) [2] also shares the spirit. ABID usually possesses collusion resistance, anonymity and attribute-privacy as well as security against impersonation. The proposed protocols in these works basically use pairing computations.

Our scheme, however, does not pursue collusion resistance, but instead, pursues less computational amount. Actually the algorithm of our scheme does not need costly pairing computation, and it is suitable for IoT devices with less computational resource.

The building blocks of our scheme are the Okamoto identification scheme [13] and the Pedersen commitment scheme [14], both of which are in the discrete logarithm setting, At a high level, the perfect witness-indistinguishability [10], [13] and the proof-of-knowledge property [5] of the Okamoto identification scheme as well as the perfectly hiding property of the Pedersen commitment scheme yield the security against concurrent attacks causing misauthentication [6] and the anonymity. On the other hand, the computationally binding property yields that the authorized person or the thing is actually a single one.

B. Organization of the Paper

In Section II, we prepare the needed notations and notions. In Section III, we provide the syntax and security definitions of our anonymous authentication scheme with decentralized multi-authorities. In Section IV, we construct a concrete scheme in the discrete logarithm setting. In Section V, We

978-1-5090-6517-2/17/$31.00 ©2017 IEEE

discuss some supplemental aspects. In Section VI, we conclude our work.

II. Preliminaries

Let λ denote the security parameter. For a positive integer n let $\mathbb{Z}^n_{>0}$ denote the set of positive integers less than or equal to n. Let $a \in_R \mathit{Space}(1^\lambda)$ denote a uniform random sampling of an element a from a sample space parametrized by 1^λ. For a set S let $|S|$ denote the number of elements of S. and for a string s let $|s|$ denote the bit-length of s. For two events A and B, let $\Pr[A : B]$ denote the conditional probability that, on condition that A occurs, B occurs. The expression $a =_? b$ returns a value 1 (True) when $a = b$, and otherwise, 0 (False). When an algorithm $\mathcal{A}$ with input a outputs z, we denote it as $z \leftarrow \mathcal{A}(a)$, or, because of space limitation, $\mathcal{A}(a) \rightarrow z$. When $\mathcal{A}$ with input a and $\mathcal{B}$ with input b interact with each other and $\mathcal{B}$ outputs z, we denote it as $z \leftarrow \langle \mathcal{A}(a), \mathcal{B}(b) \rangle$. When $\mathcal{A}$ has oracle-access to $\mathcal{O}$, we denote it as $\mathcal{A}^{\mathcal{O}}$. When $\mathcal{A}$ has concurrent oracle-access to n oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$, we denote it as $\mathcal{A}^{\mathcal{O}_i|_{i=1}^n}$. Here "concurrent" means that $\mathcal{A}$ accesses oracles in arbitrarily interleaved order of messages.

A. Discrete Logarithm Assumption [13]

Let $\mathbb{G}$ denote a cyclic group of a prime order p where $|p| = \lambda$, and let g denote a generator. Let $\mathtt{Grp}$ denote a PPT algorithm which, on input 1^λ, returns a set of public parameters $\mathtt{params} := (p, \mathbb{G}, g)$. The discrete logarithm assumption [13] states that, for any PPT algorithm $\mathcal{S}$, the advantage of $\mathcal{S}$ over $\mathtt{Grp}$, which is defined as follows, is a negligible function of λ.

$$\begin{aligned}\mathbf{Adv}^{\mathrm{dl}}_{\mathtt{Grp},\mathcal{S}}(\lambda) := \Pr[&\mathtt{params} \leftarrow \mathtt{Grp}(1^\lambda), \gamma \in_R \mathbb{Z}_p, \\ &\gamma^* \leftarrow \mathcal{S}(\mathtt{params}, g, g^\gamma) : \gamma = \gamma^*].\end{aligned}$$

Here the probability is taken over the random tape of $\mathtt{Grp}$, the uniform random sampling of γ and the random tape of $\mathcal{S}$.

B. Okamoto Identification Scheme [13]

The Okamoto identification scheme [13] is an identification scheme in the discrete logarithm setting. It is an interactive proof system whose protocol is a *Σ-protocol* [8]. Let $\mathtt{params}$ be $(p, \mathbb{G}, g)$ where p is a prime of bit-length λ, $\mathbb{G}$ is a cyclic group of order p and g is a base of $\mathbb{G}$. Choose $\alpha \in_R \mathbb{Z}_p$ and computes $h := g^\alpha$. Choose $w_1, w_2 \in_R \mathbb{Z}_p$ and computes

$$X = g^{w_1} h^{w_2}. \tag{1}$$

Set $x := (g, h, X)$ and $w := (w_1, w_2)$. The prover $\mathcal{P}$ takes as input (x, w) and the verifier $\mathcal{V}$ takes as input x. $\mathcal{P}$ chooses $\mu_1, \mu_2 \in_R \mathbb{Z}_p$ and computes a group element $A := g^{\mu_1} h^{\mu_2}$ called a commitment message. $\mathcal{P}$ sends A to $\mathcal{V}$. Then $\mathcal{V}$ chooses an exponent $c \in_R \mathbb{Z}_p$ called a challenge message. $\mathcal{V}$ sends c to $\mathcal{P}$. Then, $\mathcal{P}$ computes a pair of exponent values $\eta_1 := \mu_1 + c w_1, \eta_2 := \mu_2 + c w_2$ called a response messages. $\mathcal{P}$ sends (η_1, η_2) to $\mathcal{V}$. Finally, $\mathcal{V}$ checks whether the following equality holds to return $\mathtt{acc}$ ("accept") or $\mathtt{rej}$ ("reject").

$$g^{\eta_1} h^{\eta_2} = A X^c. \tag{2}$$

There are two features. One is that the Okamoto identification scheme is *perfectly witness-indistinguishable* [10], [13]. The verifier $\mathcal{V}$ gets no information through the protocol which witness $w = (w_1, w_2)$ satisfying the equation (1) was actually used among p solutions. The other is that the Okamoto identification scheme is a *proof of knowledge* system [5], [6]. For any PPT prover $\mathcal{P}^*$ whose probability of being accepted is non-negligible, there exist a PPT algorithm $\mathcal{K}$ called a knowledge extractor which returns a solution of the equation (1). The knowledge extractor $\mathcal{K}$ of the Okamoto identification scheme (and, generally, of the Σ-protocol) is constructed by the rewinding technique [6] to yield the special-soundness. Hence, we denote the Okamoto identification scheme as $\mathtt{OkamWIPoK}$, which will be a building block of our anonymous authentication scheme.

C. Pedersen Commitment Scheme [14]

The Pedersen commitment scheme [14] $\mathtt{PedCom}$ is a commitment scheme in the discrete logarithm setting. Let $\mathtt{params}$ be $(p, \mathbb{G}, g)$ where p is a prime of bit-length λ, $\mathbb{G}$ is a cyclic group of order p and g is a base of $\mathbb{G}$. Choose $\alpha \in_R \mathbb{Z}_p$ and computes $h := g^\alpha$. In the commitment phase, Alice, who is going to commit to an exponent value τ, chooses $w \in_R \mathbb{Z}_p$ and compute $X = g^\tau h^w$. The commitment value is X. Alice sends X to Bob. In the reveal phase, Alice sends (τ, w) to Bob and Bob checks whether X is equal to $g^\tau h^w$ or not. There are two features. One is that $\mathtt{PedCom}$ is *perfectly hiding*. Bob gets no information about the exponent expression of X to the base (g, h). The other is that $\mathtt{PedCom}$ is *computationally binding*. If (a PPT algorithm) Alice reveals in a different way $(\tau', w') \neq (\tau, w)$ with non-negligible probability, then a PPT algorithm $\mathcal{S}$ that solves the discrete logarithm problem with non-negligible probability is constructed by employing Alice as a subroutine. $\mathtt{PedCom}$ will be a building block of our anonymous authentication scheme.

III. Syntax and Security Definition

In this section, we provide the syntax and the security definition of an anonymous authentication scheme with decentralized multi-authorities, $\mathtt{a\text{-}auth}$.

A. Syntax

$\mathtt{a\text{-}auth}$ consists of five PPT algorithms, (**Setup**, **AuthKey**, **PrivKey**, $\mathcal{P}$, $\mathcal{V}$).

Setup$(1^\lambda) \rightarrow \mathtt{params}$. This PPT algorithm is needed only for generating a set of values of public parameters $\mathtt{params}$ (that includes, for example, group-operation description). On input the security parameter 1^λ, it generates the values of public parameters and returns them as $\mathtt{params}$. In the case that an already known set of values is available (for example, NIST FIPS 186-4 [12]), $\mathtt{params}$ can be set as those values.

AuthKey$(\mathtt{params}, l) \rightarrow (\mathrm{PK}_l, \mathrm{MSK}_l)$. This PPT algorithm is executed by an authority indexed by a positive integer l, who issues a private key for a prover. On input the public parameter values $\mathtt{params}$ and the index l, it generates the l-th maser public key PK_l of the authority and the corresponding

l-th master secret key MSK_l of the authority. It returns $(\mathrm{PK}_l, \mathrm{MSK}_l)$.

$\mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau) \to \mathrm{sk}_{\tau,l}$. This PPT algorithm is executed by an authority who has the l-th master secret key MSK_l. On input the public parameter values $\mathtt{params}$, the l-th pair of the public and the master secret keys $(\mathrm{PK}_l, \mathrm{MSK}_l)$ and a private identity string τ of a prover, it generates the private secret key $\mathrm{sk}_{\tau,l}$ of a prover. It returns $\mathrm{sk}_{\tau,l}$.

$\langle \mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau,l})_{l\in S}), \mathcal{V}(\mathtt{params}, (\mathrm{PK}_l)_{l\in S}) \rangle \to \mathtt{acc}/\mathtt{rej}$. These two interactive PPT algorithms are executed by a prover who is to be authenticated, and by a verifier who confirms that the prover certainly knows the secret keys issued by authorities, respectively. The prover is identified by a private string τ. Let S denote a subset of the set of all indices at which the prover is issued her private secret keys by authorities: $S := \{l \in \mathbb{Z}; \mathrm{sk}_{\tau,l} \text{ has been issued }\}$. The prover algorithm $\mathcal{P}$ and the verifier algorithm $\mathcal{V}$ are given as input the public parameter values $\mathtt{params}$ and the public keys $(\mathrm{PK}_l)_{l\in S}$. In addition, $\mathcal{P}$ is given as input the private secret keys indexed by $l \in S$, $(\mathrm{sk}_{\tau,l})_{l\in S}$. In the interaction, after at most polynomially many (in λ) moves of messages between $\mathcal{P}$ and $\mathcal{V}$, $\mathcal{V}$ returns $\mathtt{acc}$ or $\mathtt{rej}$.

B. Security Definition

For an anonymous authentication scheme with decentralized multi-authorities, $\mathtt{a\text{-}auth}$, we define two security notions; security against misauthentication and anonymity property.

1) Anonymity: For defining the anonymity of $\mathtt{a\text{-}auth}$, we consider the following experiment on $\mathtt{a\text{-}auth}$ and an adversary $\mathcal{A}$.

```
$\mathbf{Expr}^{\text{anonym}}_{\text{a-auth},\mathcal{A}}(1^\lambda, \text{poly}(\cdot))$
  $\mathtt{params} \leftarrow \mathbf{Setup}(1^\lambda), n_a \in_R \mathbb{Z}^{\text{poly}(\lambda)}_{>0}$
  For $1 \le l \le n_a : (\mathrm{PK}_l, \mathrm{MSK}_l) \leftarrow \mathbf{AuthKey}(\mathtt{params}, l)$
  $\tau_0, \tau_1 \in_R \{0,1\}^\lambda$
  For $1 \le l \le n_a :$
    $\mathrm{sk}_{\tau_0,l} \leftarrow \mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau_0)$
    $\mathrm{sk}_{\tau_1,l} \leftarrow \mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau_1)$
  $b \in_R \{0,1\}, b^* \leftarrow$
    $\langle \mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau_b,l})_{l=1}^{n_a}), \mathcal{A}(\mathtt{params}, (\mathrm{PK}_l)_{l=1}^{n_a}) \rangle$
  If $b = b^*$, then Return Win, else Return Lose
```

The advantage of an adversary $\mathcal{A}$ over an authentication scheme $\mathtt{a\text{-}auth}$ in the experiment of anonymity is defined as follows: $\mathbf{Adv}^{\text{anonym}}_{\text{a-auth},\mathcal{A}}(\lambda) \overset{\text{def}}{=} \Pr[\mathbf{Expr}^{\text{anonym}}_{\text{a-auth},\mathcal{A}}(1^\lambda, \text{poly}(\cdot)) = \mathtt{Win}]$. An authentication scheme $\mathtt{a\text{-}auth}$ is called to possess anonymity if, for any PPT algorithm $\mathcal{A}$, the advantage $\mathbf{Adv}^{\text{anonym}}_{\text{a-auth},\mathcal{A}}(\lambda)$ is negligible in λ.

2) Concurrent Attack of Misauthentication: For defining the security of $\mathtt{a\text{-}auth}$ against misauthentication, we consider concurrent attacks in the sense that an adversary $\mathcal{A}$ interacts with provers in arbitrarily interleaved order of messages in the learning phase. After the phase, $\mathcal{A}$ tries to cause misauthentication. For a formal treatment we consider the following experiment of a concurrent attack of misauthentication.

```
$\mathbf{Expr}^{\text{misauth-ca}}_{\text{a-auth},\mathcal{A}}(1^\lambda, \text{poly}(\cdot))$
  $\mathtt{params} \leftarrow \mathbf{Setup}(1^\lambda), n_a, n_p \in_R \mathbb{Z}^{\text{poly}(\lambda)}_{>0}$
  For $1 \le l \le n_a : (\mathrm{PK}_l, \mathrm{MSK}_l) \leftarrow \mathbf{AuthKey}(\mathtt{params}, l)$
  For $1 \le m \le n_p : \tau_m \in_R \{0,1\}^\lambda$,
    For $1 \le l \le n_a :$
      $\mathrm{sk}_{\tau_m,l} \leftarrow \mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau_m)$
  $st \leftarrow \mathcal{A}^{\mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau_m,l})_{l=1}^{n_a})|_{m=1}^{n_p}}(\mathtt{params}, (\mathrm{PK}_l)_{l=1}^{n_a})$
  $d \leftarrow \langle \mathcal{A}(st), \mathcal{V}(\mathtt{params}, (\mathrm{PK}_l)_{l=1}^{n_a}) \rangle$
  If $d = \mathtt{acc}$, then Return Win, else Return Lose
```

In the above experiment, $\mathcal{A}$ does concurrent oracle accesses to $\mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau_m,l})_{l=1}^{n_a})$, $m = 1, \dots, n_p$, as the learning phase. Then, as the next phase, $\mathcal{A}$ interacts with $\mathcal{V}$. In the interaction $\mathcal{A}$ generates a set of indices S which determines public keys $\{\mathrm{PK}_l; l \in S\}$. Under the set of public keys $\mathcal{A}$ tries to cause misauthentication.

The advantage of an adversary $\mathcal{A}$ over an authentication scheme $\mathtt{a\text{-}auth}$ in the experiment is defined as follows: $\mathbf{Adv}^{\text{misauth-ca}}_{\text{a-auth},\mathcal{A}}(\lambda) \overset{\text{def}}{=} \Pr[\mathbf{Expr}^{\text{misauth-ca}}_{\text{a-auth},\mathcal{A}}(1^\lambda, \text{poly}(\cdot)) = \mathtt{Win}]$. An authentication scheme $\mathtt{a\text{-}auth}$ is called secure against concurrent attacks of misauthentication if, for any PPT algorithm $\mathcal{A}$, the advantage $\mathbf{Adv}^{\text{misauth-ca}}_{\text{a-auth},\mathcal{A}}(\lambda)$ is negligible in λ.

IV. Our Construction

In this section, we construct a concrete anonymous authentication scheme with decentralized multi-authorities, $\mathtt{a\text{-}auth} = (\mathbf{Setup}, \mathbf{AuthKey}, \mathbf{PrivKey}, \mathcal{P}, \mathcal{V})$. Then we prove that our $\mathtt{a\text{-}auth}$ possesses anonymity and that our $\mathtt{a\text{-}auth}$ is secure against concurrent attacks of misauthentication, both under the discrete logarithm assumption on the group generation algorithm, $\mathtt{Grp}$.

A. Scheme

The five algorithms are constructed as follows (see Fig. 1).

$\mathbf{Setup}(1^\lambda) \to \mathtt{params}$. On input the security parameter 1^λ, this PPT algorithm runs $\mathtt{Grp}(1^\lambda)$ to generates a prime p, a cyclic group $\mathbb{G}$ of order p and a base g_0 of $\mathbb{G}$. Besides, it chooses an exponent $\alpha_0 \in_R \mathbb{Z}_p$ and computes $h_0 := g_0^{\alpha_0}$. It returns $\mathtt{params} := (p, \mathbb{G}, g_0, h_0)$.

$\mathbf{AuthKey}(\mathtt{params}, l) \to (\mathrm{PK}_l, \mathrm{MSK}_l)$. On input $\mathtt{params}$ and the index l, this PPT algorithm chooses a base $g_l \in_R \mathbb{G}$ and an exponent $\alpha_l \in_R \mathbb{Z}_p$ and computes $h_l := g_l^{\alpha_l}$. Then it chooses two exponents $\tau_l^*, w_l^* \in_R \mathbb{Z}_p$ and computes $X_l := g_l^{\tau_l^*} h_l^{w_l^*}$. It sets $\mathrm{PK}_l := (g_l, h_l, X_l), \mathrm{MSK}_l := (\alpha_l, \tau_l^*, w_l^*)$. It returns $(\mathrm{PK}_l, \mathrm{MSK}_l)$.

$\mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau) \to \mathrm{sk}_{\tau,l}$. On input $\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l$ and a private identity string τ of a prover, this PPT algorithm computes $w_{\tau,l} := w_l^* + (\tau_l^* - \tau)/\alpha_l$ and sets $\mathrm{sk}_{\tau,l} := (\tau, w_{\tau,l})$. It returns $\mathrm{sk}_{\tau,l}$. Note here that the following equality holds.

$$X_l = g_l^{\tau_l^*} h_l^{w_l^*} = g_l^{\tau} h_l^{w_{\tau,l}}. \tag{3}$$

$\langle \mathcal{P}(\texttt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau,l})_{l\in S}), \mathcal{V}(\texttt{params}, (\mathrm{PK}_l)_{l\in S})\rangle \rightarrow$ $\texttt{acc}/\texttt{rej}$. On the common input $\texttt{params}$ and $(\mathrm{PK}_l)_{l\in S}$ and the private input $(\mathrm{sk}_{\tau,l})_{l\in S}$, $\mathcal{P}$ chooses an exponent $w_{\tau,0} \in_R \mathbb{Z}_p$ and computes the Perdersen commitment $X_0 := g_0^{\tau} h_0^{w_{\tau,0}}$ to τ. Then, for each $l \in S$, $\mathcal{P}$ chooses two exponents $\mu_l, \nu_l \in_R \mathbb{Z}_p$ and computes the commitment message $A_l := g_l^{\mu_l} h_l^{\nu_l}$ of the Σ-protocol of $\texttt{OkamWIPoK}$. Besides, $\mathcal{P}$ chooses an exponent $\nu_{0,l} \in_R \mathbb{Z}_p$ and computes the commitment message $A_{0,l} := g_0^{\mu_l} h_0^{\nu_{0,l}}$ of $\texttt{OkamWIPoK}$. $\mathcal{P}$ sends X_0 and $(A_l, A_{0,l})_{l\in S}$ to $\mathcal{V}$.

Receiving the Pedersen commitment X_0 and all the commitment messages $(A_l, A_{0,l})_{l\in S}$ of $\texttt{OkamWIPoK}$, $\mathcal{V}$ on input $\texttt{params}$ and $(\mathrm{PK}_l)_{l\in S}$ chooses a challenge message $c \in_R \textsc{ChaSp}(1^{\lambda})$ of the Σ-protocol of $\texttt{OkamWIPoK}$. Here $\textsc{ChaSp}(1^{\lambda})$ is $\mathbb{Z}_p$. $\mathcal{V}$ sends c to $\mathcal{P}$.

Receiving the challenge message c, $\mathcal{P}$ computes the response messages of the Σ-protocol of $\texttt{OkamWIPoK}$ as $\eta_l := \mu_l + c\tau$, $\theta_l := \nu_l + cw_{\tau,l}$ and $\theta_{0,l} := \nu_{0,l} + cw_{\tau,0}$. $\mathcal{P}$ sends $(\eta_l, \theta_l, \theta_{0,l})_{l\in S}$ to $\mathcal{V}$.

Receiving the response messages $(\eta_l, \theta_l, \theta_{0,l})_{l\in S}$ of $\texttt{OkamWIPoK}$, $\mathcal{V}$ checks the following equations of the Σ-protocol of $\texttt{OkamWIPoK}$: For $l \in S$, $g_l^{\eta_l} h_l^{\theta_l} =_? A_l X_l^c$ and $g_0^{\eta_l} h_0^{\theta_{0,l}} =_? A_{0,l} X_0^c$. If all the equations hold, then $\mathcal{V}$ returns $\texttt{acc}$, and otherwise, $\texttt{rej}$.

The correctness holds for our $\texttt{a-auth}$. That is, for any given positive polynomial $\mathrm{poly}(\lambda)$,

$$\begin{aligned}
&\Pr[\texttt{params} \leftarrow \mathbf{Setup}(1^{\lambda}); n_a \in_R \mathbb{Z}_{>0}^{\mathrm{poly}}(\lambda);\\
&\text{For } l = 1 \text{ to } n_a\ (\mathrm{PK}_l, \mathrm{MSK}_l) \leftarrow \mathbf{AuthKey}(\texttt{params}, l);\\
&\tau \in_R \{0,1\}^{\lambda};\\
&\text{For } l = 1 \text{ to } n_a\ \mathrm{sk}_{\tau,l} \leftarrow \mathbf{PrivKey}(\texttt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau);\\
&S := \mathbb{Z}_{>0}^{n_a} :\\
&\text{acc} \leftarrow \langle \mathcal{P}(\texttt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau,l})_{l\in S}), \mathcal{V}(\texttt{params}, (\mathrm{PK}_l)_{l\in S})\rangle]\\
&= 1. \qquad (4)
\end{aligned}$$

B. Security

1) Proofs of Knowledge of Common Prefix:

Theorem 1 (PoK of Common String). *Suppose that $|S| \geq 2$. Then, our authentication scheme $\texttt{a-auth}$ is a proof of knowledge system for generating proofs of the knowledge of witnesses that share a common string τ under the discrete logarithm assumption on $\texttt{Grp}$. More precisely, given $\texttt{params}$ and $(PK_l)_{l\in S}$ and for any PPT prover $\mathcal{P}^*$ whose probability of being accepted is p_{acc}, there exist a PPT knowledge extractor $\mathcal{K}$ and a PPT algorithm $\mathcal{S}$ on $\texttt{Grp}$ where $\mathcal{K}(\texttt{params}, (PK_l)_{l\in S})$ returns a tuple $(\hat{\tau}, (\hat{w_l}, \hat{w_{0,l}})_{l\in S}) \in (\mathbb{Z}_p)^{2|S|+1}$ that satisfies the following equations:*

$$\begin{cases} X_l &= g_l^{\hat{\tau}} h_l^{\hat{w_l}}, \\ X_0 &= g_0^{\hat{\tau}} h_0^{\hat{w_{0,l}}}. \end{cases} l \in S, \qquad (5)$$

with the probability at least

$$\left(p_{acc} - \frac{1}{p}\right)^2 - \mathbf{Adv}^{dl}_{Grp,\mathcal{S}}(\lambda). \qquad (6)$$

Proof. Using a PPT prover $\mathcal{P}^*$ as a subroutine, we construct a PPT knowledge extractor $\mathcal{K}$ as follows (in accordance with the standard argument [6]). $\mathcal{K}$ is given as input $(\texttt{params}, (\mathrm{PK}_l)_{l\in S})$. $\mathcal{K}$ gives $(\texttt{params}, (\mathrm{PK}_l)_{l\in S})$ to $\mathcal{P}^*$ and receives from $\mathcal{P}^*$ a tuple of commitment messages $(A_l, A_{0,l})_{l\in S}$ and copies and keeps the inner state *st* of $\mathcal{P}^*$. $\mathcal{K}$ chooses two challenge messages c and c' uniformly at random from $\textsc{ChaSp}(\lambda) = \mathbb{Z}_p$. Here $c = c'$ holds with probability $1/p$. $\mathcal{K}$ sends c to $\mathcal{P}^*(st)$ and receives the response messages $(\eta_l, \theta_l, \theta_{0,l})_{l\in S}$. $\mathcal{K}$ also sends c' to $\mathcal{P}^*(st)$ and receives the response messages $(\eta'_l, \theta'_l, \theta'_{0,l})_{l\in S}$. Suppose here that all the messages are accepting conversation. For each $l \in S$, $\mathcal{K}$ computes the differences: $\Delta\eta_l := \eta_l - \eta'_l, \Delta\theta_l := \theta_l - \theta'_l, \Delta\theta_{0,l} := \theta_{0,l} - \theta'_{0,l}, \Delta c := c - c'$. After cancelling the commitment messages the following equalities hold.

$$\begin{cases} g_l^{\Delta\eta_l} h_l^{\Delta\theta_l} &= X_l^{\Delta c}, \\ g_0^{\Delta\eta_l} h_0^{\Delta\theta_{0,l}} &= X_0^{\Delta c}, \end{cases} l \in S. \qquad (7)$$

As $\Delta c \neq 0$ except the case of probability $1/p$, $\mathcal{K}$ sets for each l $\hat{\tau_l} := \Delta\eta_l/\Delta c, \hat{w_l} := \Delta\theta_l/\Delta c$ and $\hat{w_{0,l}} := \Delta\theta_{0,l}/\Delta c$. Then the following equalities hold.

$$\begin{cases} X_l &= g_l^{\hat{\tau_l}} h_l^{\hat{w_l}}, \\ X_0 &= g_0^{\hat{\tau_l}} h_0^{\hat{w_{0,l}}}, \end{cases} l \in S. \qquad (8)$$

Thus, $\mathcal{K}$ is able to compute the tuple $(\hat{\tau_l}, \hat{w_l}, \hat{w_{0,l}})_{l\in S}$ that satisfies the equation (8).

Claim 1. *The equality*

$$\exists \hat{\tau} \in \mathbb{Z}_p \textit{ s.t. } [\forall l \in S, \hat{\tau_l} = \hat{\tau}] \qquad (9)$$

holds except the case of probability at most $\mathbf{Adv}^{dl}_{Grp,\mathcal{S}}(\lambda)$.

Proof of Claim 1. Name the event that there exists at least a pair (l_1, l_2) at which $\hat{\tau_{l_1}} \neq \hat{\tau_{l_2}}$ as $\textsc{DlPair}$. A solver $\mathcal{S}$ of the discrete logarithm problem on $\texttt{Grp}$ is constructed, as follows. Given an instance (g, h), $\mathcal{S}$ sets $g_0 := g$ and $h_0 := h$. $\mathcal{S}$ executes $\mathbf{Setup}, \mathbf{AuthKey}, \mathbf{PrivKey}$ honestly as in the correctness formula (4). Then, $\mathcal{S}$ interacts with $\mathcal{P}^*(\texttt{params}, (\mathrm{PK}_l)_{l\in S})$ as a verifier, and obtains all the above messages (as in the proof of the theorem).

We are in the case that $|S| \geq 2$. Hence, if $\hat{\tau_{l_1}} \neq \hat{\tau_{l_2}}$ in (8) for some $l_1, l_2 \in S, l_1 \neq l_2$, then $\mathcal{S}$ obtains two different expressions of the Pedersen commitment X_0 to the base (g_0, h_0). Therefore $\mathcal{S}$ computes the discrete log of h to the base g as $\log_g(h) = -(\hat{\tau_{l_1}} - \hat{\tau_{l_2}})/(\hat{w_{0,l_1}} - \hat{w_{0,l_2}})$. The probability of this event is $\Pr[\textsc{DlPair}]$, which is less than or equal to $\mathbf{Adv}^{\mathrm{dl}}_{\texttt{Grp},\mathcal{S}}(\lambda)$ by the definition. □

Now we evaluate the probability in the theorem. Using Reset Lemma [6] (Lemma 3.1), the following inequality holds.

$$p_{acc} \leq (1/p) + \sqrt{\Pr[\mathcal{K} \text{ returns } (\hat{\tau_l}, \hat{w_l}, \hat{w_{0,l}})_{l\in S}]} \qquad (10)$$

By the above Claim 1, the following equality holds.

$$\begin{aligned} &\Pr[\mathcal{K} \text{ returns } (\hat{\tau_l}, \hat{w_l}, \hat{w_{0,l}})_{l\in S}] - \mathbf{Adv}^{\mathrm{dl}}_{\texttt{Grp},\mathcal{S}}(\lambda) \\ &\leq \Pr[\mathcal{K} \text{ returns } (\hat{\tau}, (\hat{w_l}, \hat{w_{0,l}})_{l\in S})]. \end{aligned} \qquad (11)$$

Combining (10) and (11), the evaluation (6) follows. □

```
$\mathbf{Setup}(1^\lambda)$
  $(p, \mathbb{G}, g_0) \leftarrow \mathtt{Grp}(1^\lambda)$
  $\alpha_0 \in_R \mathbb{Z}_p, h_0 := g_0^{\alpha}$
  $\mathtt{params} := (p, \mathbb{G}, g_0, h_0)$
  Return $\mathtt{params}$

$\mathbf{AuthKey}(\mathtt{params}, l)$
  $g_l \in_R \mathbb{G}, \alpha_l \in_R \mathbb{Z}_p, h_l := g_l^{\alpha_l}$
  $\tau_l^*, w_l^* \in_R \mathbb{Z}_p, X_l := g_l^{\tau_l^*} h_l^{w_l^*}$
  $\mathrm{PK}_l := (g_l, h_l, X_l)$
  $\mathrm{MSK}_l := (\alpha_l, \tau_l^*, w_l^*)$
  Return $(\mathrm{PK}_l, \mathrm{MSK}_l)$

$\mathbf{PrivKey}(\mathtt{params}, \mathrm{PK}_l, \mathrm{MSK}_l, \tau)$
  $w_{\tau,l} := w_l^* + (\tau_l^* - \tau)/\alpha_l$
  $\mathrm{sk}_{\tau,l} := (\tau, w_{\tau,l})$, Return $\mathrm{sk}_{\tau,l}$

$\mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau,l})_{l \in S})$                              $\mathcal{V}(\mathtt{params}, (\mathrm{PK}_l)_{l \in S})$
  $w_{\tau,0} \in_R \mathbb{Z}_p, X_0 := g_0^{\tau} h_0^{w_{\tau,0}}$
  For $l \in S$ :
    $\mu_l, \nu_l \in_R \mathbb{Z}_p, A_l := g_l^{\mu_l} h_l^{\nu_l}$
    $\nu_{0,l} \in_R \mathbb{Z}_p, A_{0,l} := g_0^{\mu_l} h_0^{\nu_{0,l}}$        $X_0, (A_l, A_{0,l})_{l \in S}$
                                                                $\longrightarrow$
                                                                                   $c \in_R \textsc{ChaSp}(1^\lambda)$
                                                                $c$
  For $l \in S$ :                                               $\longleftarrow$
    $\eta_l := \mu_l + c\tau, \theta_l := \nu_l + cw_{\tau,l}$
                 $\theta_{0,l} := \nu_{0,l} + cw_{\tau,0}$        $(\eta_l, \theta_l, \theta_{0,l})_{l \in S}$
                                                                $\longrightarrow$          For $l \in S$ :
                                                                                     $g_l^{\eta_l} h_l^{\theta_l} =_? A_l X_l^c$
                                                                                     $g_0^{\eta_l} h_0^{\theta_{0,l}} =_? A_{0,l} X_0^c$
                                                                                   If all eqs. hold, Return $\mathtt{acc}$
                                                                                   otherwise Return $\mathtt{rej}$
```

Fig. 1. Anonymous authentication scheme with decentralized multi-authorities a-auth

2) Anonymity:

Theorem 2 (Perfect Anonymity). *Our authentication scheme a-auth possesses perfect anonymity. More precisely, for any unbounded algorithm $\mathcal{A}$, the following inequality holds.*

$$\mathbf{Adv}^{\textit{anonym}}_{\textit{a-auth},\mathcal{A}}(\lambda) = 0.$$

Proof (sketch). This is the direct consequence of two facts; one is that the Okamoto identification scheme [13] is perfectly witness indistinguishable; the other is that the Pedersen commitment scheme [14] is perfectly hiding. □

3) Security against Concurrent Attack of Misauthentication:

Theorem 3 (Security under Concurrent Attacks of Causing Misauthentication). *Our authentication scheme a-auth is secure against concurrent attacks of causing misauthentication under the discrete logarithm assumption on Grp. More precisely, given a positive polynomial poly$(\cdot)$ and for any PPT algorithm $\mathcal{A}$ in the experiment of the concurrent attack, there exists a PPT algorithm $\mathcal{S}$ on Grp that satisfies the following inequality.*

$$\mathbf{Adv}^{\textit{misauth-ca}}_{\textit{a-auth},\mathcal{A}}(\lambda) \leq \frac{1}{p} + \sqrt{\mathbf{Adv}^{\textit{dl}}_{\textit{Grp},\mathcal{S}}(\lambda) + \frac{1}{p}}. \tag{12}$$

Proof (sketch). Using a PPT algorithm $\mathcal{A}$ as a subroutine, we construct a PPT solver $\mathcal{S}$ of the discrete logarithm problem on $\mathtt{Grp}$ as follows. Given an instance (g, h), $\mathcal{S}$ runs $\mathbf{Setup}(1^\lambda)$ and chooses n_a, n_p as in the experiment $\mathbf{Expr}^{\text{misauth-ca}}_{\text{a-auth},\mathcal{A}}(1^\lambda, \text{poly}(\cdot))$. $\mathcal{S}$ sets $g_0 := g$ and $h_0 := h$. Instead of honestly running $\mathbf{AuthKey}$ and $\mathbf{PrivKey}$, $\mathcal{S}$ randomizes g and h to obtain g_l and h_l for $l = 1, \ldots, n_a$, as follows. For each $l = 1, \ldots, n_a$, $\mathcal{S}$ chooses $\beta_l, \gamma_l \in_R \mathbb{Z}_p$ and computes $g_l := g^{\beta_l}, h_l := h^{\gamma_l}$. Further, $\mathcal{S}$ chooses a single $\tau^* \in_R \mathbb{Z}_p$. Then for each $l = 1, \ldots, n_a$, $\mathcal{S}$ chooses $w_l^* \in_R \mathbb{Z}_p$ and computes $X_l := g_l^{\tau^*} h_l^{w_l^*}$. The public key PK_l is (g_l, h_l, X_l).

Using the values $(\tau^*, (w_l^*)_{l=1}^{n_a})$ instead of private secret keys $((\mathrm{sk}_{\tau_m,l})_{l=1}^{n_a})_{m=1}^{n_p}$ that are not generated, $\mathcal{S}$ simulates $\mathcal{P}(\mathtt{params}, (\mathrm{PK}_l, \mathrm{sk}_{\tau_m,l})_{l=1}^{n_a})|_{m=1}^{n_p}$ perfectly in the replies to the concurrent oracle access by $\mathcal{A}$. This can be done because of the perfect anonymity of a-auth (Theorem 2); $\mathcal{A}$ learns no information about the exponent expression (3) of X_l to the base (g_l, h_l), $l = 1, \ldots, n_a$. $\mathcal{S}$ copies and keeps the inner state *st* of $\mathcal{A}$. After the concurrent oracle access, $\mathcal{A}$ interacts with a verifier $\mathcal{V}(\mathtt{params}, (\mathrm{PK}_l)_{l=1}^{n_a})$ simulated by $\mathcal{S}$. $\mathcal{S}$ rewinds $\mathcal{A}(\textit{st})$ to obtain the equalities that is the same as (8) (in the proof of Theorem 1). Therefore, there exists at least one l at which the following equalities hold.

$$X_l = g_l^{\tau^*} h_l^{w_l^*} = g_l^{\hat{\tau}_l} h_l^{\hat{w}_l}. \tag{13}$$

Since $\mathcal{A}$ learns no information about the exponent expression, $(\tau^*, w_l^*) \neq (\hat{\tau}_l, \hat{w}_l)$ holds without the probability at most $1/p$. Thus, $\mathcal{S}$ is able to compute the discrete logarithm as $\log_{g_l}(h_l) = -(\tau^* - \hat{\tau}_l)/(w_l^* - \hat{w}_l)$ and hence, $\log_g(h) = -(\beta_l/\gamma_l)(\tau^* - \hat{\tau}_l)/(w_l^* - \hat{w}_l)$, without the probability at most $1/p$. The probability evaluation (12) follows from Reset Lemma [6] (Lemma 3.1). □

V. Discussion

In this section, we discuss the security proof of Theorem 3, a generalization of our proof system to monotone predicates, and the drawback of vulnerability against collusion attacks.

A. Security Proof Revisited

Roughly speaking, the two properties, the witness-indistinguishability (WI) and the proof-of-knowledge property (PoK), yield the security against concurrent attacks (see [6]). Our a-auth has WI and PoK, and therefore the concurrent security follows.

B. Generalization to any Monotone Predicates

Our a-auth in this paper is for proving the knowledge of the exponents which satisfy all the public equations. By using the OR-proof [9], a-auth is used for proving the knowledge of the exponents which satisfy *at least one* of the public equations. The critical change in the protocol of a-auth is to divide the challenge message c into $c = c_1 \oplus c_2$, where one of c_1 and c_2 is chosen at random. By using an extended division technique [3], [4], a-auth is generalized to a system for proving the knowledge of witnesses that satisfy any given monotone predicate.

C. Collusion Attacks

Our a-auth has a drawback that a-auth is *not* secure against collusion attacks. That is, if two secret keys $\mathrm{sk}_{\tau_1,l} = (\tau_1, w_{\tau_1,l})$ and $\mathrm{sk}_{\tau_2,l} = (\tau_2, w_{\tau_2,l})$ issued by a single l-th authority are collected, then the core component α_l of the master secret key MSK_l is computed as $\alpha_l = -(\tau_1 - \tau_2)/(w_{\tau_1,l} - w_{\tau_2,l})$. To avoid the vulnerability, there are two possible countermeasures by real operations. One is for the authority to update the core component α_l, and reissue the private secret keys to provers. The other is to make a premise that collecting private identity strings $\{\tau_m\}$ causes some critical issue like in the case as the social security numbers in USA.

VI. Conclusion

We proposed an anonymous authentication scheme a-auth with a feature that various authorities can admit a person or a thing without a central authority. The building blocks of our scheme are the Okamoto identification scheme OkamWIPoK and the Pedersen commitment scheme PedCom. The witness-indistinguishability and the proof-of-knowledge property of OkamWIPoK and the perfect hiding property of PedCom gave the anonymity and concurrent security of a-auth. On the other hand, the binding property of PedCom enabled to prove the knowledge of witnesses that share a common identity string. a-auth does not need costly pairing computation, and hence our scheme is suitable for IoT devices with less computational resource.

It is expected that we can exit the drawback of vulnerability against collusion attacks by employing (and paying the cost of) the pairing technique of the Groth-Sahai non-interactive witness-indistinguishable proof system [11] and commitments to group elements [1]. We leave the construction as our future work.

Acknowledgment

This work is partially supported by kakenhi Grant-in-Aid for Scientific Research (C) JP15K00029 from Japan Society for the Promotion of Science.

References

[1] M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo, “Structure-preserving signatures and commitments to group elements,” in *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, 2010, pp. 209–236. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-14623-7_12

[2] H. Anada, S. Arita, S. Handa, and Y. Iwabuchi, “Attribute-based identification: Definitions and efficient constructions,” in *Information Security and Privacy - 18th Australasian Conference, ACISP 2013, Brisbane, Australia, July 1-3, 2013. Proceedings*, 2013, pp. 168–186. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-39059-3_12

[3] H. Anada, S. Arita, and K. Sakurai, “Attribute-based signatures without pairings via the fiat-shamir paradigm,” in *ASIAPKC2014*, ser. ACM-ASIAPKC, vol. 2. ACM, 2014, pp. 49–58.

[4] ——, “Proofs of knowledge on monotone predicates and its application to attribute-based identifications and signatures,” *IACR Cryptology ePrint Archive*, vol. 2016, p. 483, 2016. [Online]. Available: http://eprint.iacr.org/2016/483

[5] M. Bellare and O. Goldreich, “On defining proofs of knowledge,” in *Advances in Cryptology - CRYPTO ’92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, 1992, pp. 390–420. [Online]. Available: http://dx.doi.org/10.1007/3-540-48071-4_28

[6] M. Bellare and A. Palacio, “GQ and Schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks,” in *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*, 2002, pp. 162–177. [Online]. Available: http://dx.doi.org/10.1007/3-540-45708-9_11

[7] J. Camenisch, M. Dubovitskaya, K. Haralambiev, and M. Kohlweiss, “Composable and modular anonymous credentials: Definitions and practical constructions,” in *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conf. on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, 2015, pp. 262–288. [Online]. Available: http://dx.doi.org/10.1007/978-3-662-48800-3_11

[8] R. Cramer, “Modular designs of secure, yet practical cyptographic protocols,” Ph.D. dissertation, University of Amsterdam, Amsterdam, the Netherlands, 1996.

[9] R. Cramer, I. Damgård, and B. Schoenmakers, “Proofs of partial knowledge and simplified design of witness hiding protocols,” in *CRYPTO ’94*. Springer-Verlag, 1994, pp. 174–187.

[10] U. Feige and A. Shamir, “Witness indistinguishable and witness hiding protocols,” in *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, 1990, pp. 416–426. [Online]. Available: http://doi.acm.org/10.1145/100216.100272

[11] J. Groth and A. Sahai, “Efficient non-interactive proof systems for bilinear groups,” in *Proceedings of the Theory and Applications of Cryptographic Techniques 27th Annual International Conference on Advances in Cryptology*, ser. EUROCRYPT’08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 415–432. [Online]. Available: http://dl.acm.org/citation.cfm?id=1788414.1788438

[12] National Institute of Standards and Technology, “Federal information processing standard (fips) 186-4, digital signature standard (dss),” 2013, http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf.

[13] T. Okamoto, “Provably secure and practical identification schemes and corresponding signature schemes,” in *Advances in Cryptology - CRYPTO ’92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, 1992, pp. 31–53. [Online]. Available: http://dx.doi.org/10.1007/3-540-48071-4_3

[14] T. P. Pedersen, “Non-interactive and information-theoretic secure verifiable secret sharing,” in *Advances in Cryptology - CRYPTO ’91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, 1991, pp. 129–140. [Online]. Available: http://dx.doi.org/10.1007/3-540-46766-1_9

[15] S. Sadiah, T. Nakanishi, and N. Funabiki, “Anonymous credential system with efficient proofs for monotone formulas on attributes,” in *Advances in Information and Computer Security - 10th International Workshop on Security, IWSEC 2015, Nara, Japan, August 26-28, 2015, Proceedings*, 2015, pp. 262–278. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-22425-1_16