# A SECURE AND PRIVACY-PROTECTING PROTOCOL FOR TRANSMITTING PERSONAL INFORMATION BETWEEN ORGANIZATIONS

*David Chaum & Jan-Hendrik Evertse*

Centre for Mathematics and Computer Science
Kruislaan 413 1098 SJ Amsterdam The Netherlands

***Abstract:*** A multi-party cryptographic protocol and a proof of its security are presented. The protocol is based on RSA using a one-way-function. Its participants are individuals and organizations, which are not assumed to trust each other. The protocol implements a "credential mechanism", which is used to transfer personal information about individuals from one organization to another, while allowing individuals to retain substantial control over such transfers.

It is proved that the privacy of individuals is protected in a way that is optimal against cooperation of all organizations, even if the organizations have infinite computational resources. We introduce a "formal credential mechanism", based on an "ideal RSA cryptosystem". It allows individuals a chance of successful cheating that is proved to be exponentially small in the amount of computation required. The new proof techniques used are based on probability theory and number theory and may be of more general applicability.

## 1. INTRODUCTION

The aim of this paper is to present in a formal way, and to prove the desired properties of, a multi-party cryptographic protocol called a "credential mechanism" that was introduced in [Ch 85]. In this section, the protocol is re-introduced and then an overview of the paper is given.

### 1.1. Credential mechanisms

A credential mechanism is a cryptographic protocol that provides for transfers of information about individuals between organizations. The information about individuals transferred will consist of *credentials* belonging to some fixed set. If individuals identify themselves to each organiza-

This research was supported in part by the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

A.M. Odlyzko (Ed.): Advances in Cryptology - CRYPTO '86, LNCS 263, pp. 118-167, 1987.

tion essentially uniquely, such as by their name, date of birth, or some universal identification number, then credentials about an individual can be transferred between organizations without control by that individual. To give individuals control over such transfers, the credential mechanism allows individuals to use different *pseudonyms* with different organizations. Different individuals use different pseudonyms. Organizations have no more identifying information about individuals than these pseudonyms, and thus, from the point of view of the organizations, credentials can be linked to pseudonyms rather than to individuals.

When information about an individual is to be sent from one organization to another, the first organization issues a certificate, called a "credential on a pseudonym", to the individual, showing that a particular credential applies to his pseudonym used with that organization; then the individual transforms this certificate into "the same credential on a (second) pseudonym" used with the second organization; and finally the individual shows this credential on the second pseudonym to the second organization.

The following is a precise description of the properties of the pseudonyms and credentials of the credential mechanism:

**Property 1.** The set of pseudonyms can be partitioned in two ways: into I-sets each containing the pseudonyms used by an individual and into O-sets each containing the pseudonyms known to an organization.

**Property 2.** Each I-set and each O-set have at most one pseudonym in common.

**Property 3.** For any individual, it is easy to compute a credential on a pseudonym if some organization has issued the same credential on a pseudonym belonging to the same I-set; otherwise computing that credential on that pseudonym is infeasible for that individual (unforgeability).

**Property 4.** The credential mechanism does not reveal any information to even cooperating organizations about how the pseudonyms are partitioned into I-sets (unlinkability).

By these properties, a credential mechanism guarantees each individual that different organizations (possibly by cooperation with other organizations and some other individuals) can never link the information they have about him. For an organization can only link the information about that individual to his pseudonym used with that organization; and by property 4 the credential mechanism does not reveal to the organizations which pseudonyms belong to which individual. A credential mechanism also protects each organization against individuals trying to convince it that certain credentials apply to them while this is in fact not true. This is so since by property 3, no individual is able to compute a credential on one of his pseudonyms if he did not previously get that credential on one of his other pseudonyms. Property 2 also protects organizations by, for example, preventing a credential issued by one organization on some pseudonym from being transformed into a credential on more than one pseudonym used with any particular organization.

To achieve properties 1-3, both pseudonyms and credentials on pseudonyms must be constructed in a special way. The credential mechanism must ensure that individuals construct their pseudonyms in this way. But, because of property 4, individuals cannot be required to show

organizations how they have constructed their pseudonyms. Therefore, credential mechanisms include a *validating part* which is a protocol by which individuals convince the organizations that they have constructed their pseudonyms correctly without revealing how they have done so.

### 1.2. Overview of the paper

In §2 we introduce some formalism about protocols and attacks on protocols (these are ways by which some of the participants of the protocol, possibly by cooperating, violate the rules of a protocol) which will serve as a mathematical framework in which properties of the credential mechanism can be stated and proved.

In §3 we describe a credential mechanism based on RSA with a single composite modulus $N$. The validating part is based on a "one-way" function, and all credentials on pseudonyms are RSA-signatures. Since only one modulus is used, and since it is not assumed that all organizations trust each other, a special organization participates in the credential mechanism, called a "signature authority", which is the only organization that has to know the factorization of $N$ used in making signatures. The signature authority is trusted by the other organizations—but not by the individuals—and is willing to provide suitable signatures requested by organizations. The signature authority also participates in the validating part. In §§3.1-3.4 an overview of the credential mechanism is given which can be read independently of §2. In §3.5 the credential mechanism is described by means of the formalism introduced in §2.

In §4 we prove that property 4 (the unlinkability of the pseudonyms) holds for the credential mechanism as described in §3.5 in the following respect: all information revealed by the credential mechanism about how the pseudonyms are partitioned into I-sets is already revealed by the *moments* that pseudonyms, credentials, etc. are issued by or shown to an organization. It is argued that this kind of information is revealed by any credential mechanism, so that our credential mechanism offers optimal unlinkability.

In §5 we introduce the "formal credential mechanism". This mechanism is equivalent to the actual credential mechanism, except that it is based on an "ideal" RSA cryptosystem and an "ideal" one-way function. It is possible to establish a correspondence between messages in "ideal RSA" and messages in "real RSA" by means of a multiplicative homomorphism. Our formal credential mechanism is endowed with a computational model which precisely describes which "computations" each participant of the credential mechanism can perform. Thus our model of a formal credential mechanism can be compared with that used for RSA based ping-pong protocols in [EGS 85].

In §6 we state the main theorem about the formal credential mechanism: that in each possible attack on the formal credential mechanism, the probability that individuals will agree with the organizations about the use of pseudonyms which do not have properties 1, 2 and 3, has an upper bound which is an exponentially decreasing function of the number of computations done in the validating part. In §6 we also give an example of an attack by which individuals could try to agree with organizations about the use of pseudonyms in the formal credential mechanism

which do not satisfy properties 1, 2 and 3 mentioned in §1.1. With this attack we show that the upper bound given in the theorem cannot essentially be improved.

In §7, we prove the theorem mentioned in §6. Unlike the ping-pong protocols of [EGS 85], our formal credential mechanism models a protocol in which RSA is used with only a single modulus, but with different encryption and decryption exponents. Therefore our method of proof is entirely different from theirs. §§5-7 can be read independently of §4.

The reason that we prove properties 1, 2 and 3 for the formal credential mechanism instead of the actual credential mechanism, is that the following seems likely: for a proper choice of the composite modulus and the one-way function, it is computationally infeasible for an individual to agree with an organization about the use of a pseudonym, if that individual is not able to agree with the organization about the use of the corresponding pseudonym in the formal credential mechanism. An investigation of the correctness of this statement is beyond the scope of this paper.

In §8 we mention a few extensions of the credential mechanism.

## 2. PROTOCOLS AND ATTACKS

For the analysis of the credential mechanism provided in this paper, it is necessary to make clear what is meant by notions like "protocols" or "attacks on protocols". In this section we give definitions of these notions which are modifications of those of DeMillo, Lynch and Merritt [DLM 82]. As noted before, this section need not be read before §§3.1-3.4 in which the credential mechanism is introduced.

### 2.1. Some probability theory

In the sequel we need some discrete probability theory which is introduced here.

We fix an enumerable set $\Omega = \{\omega_1, \omega_2, \cdots\}$. To each $\omega_i$ in $\Omega$ we attach a real number $Pr[\omega_i]$ in the closed interval $[0,1]$ such that

$$\sum_{i=1}^{\infty} Pr[\omega_i] = 1 .$$

Subsets of $\Omega$ are called *events*. The *probability* of event $\mathcal{A}$, denoted by $Pr[\mathcal{A}]$, is given by

$$\sum_{\omega \in \mathcal{A}} Pr[\omega] .$$

The *conditional probability* of event $\mathcal{A}$, given event $\mathcal{B}$, is defined by

$$Pr[\mathcal{A} \mid \mathcal{B}] = \frac{Pr[\mathcal{A} \cap \mathcal{B}]}{Pr[\mathcal{B}]}$$

if $Pr[\mathfrak{B}]\neq 0$ and is not defined otherwise. When stating results involving conditional probabilities we always assume that these are defined, without explicitly mentioning this. If $\mathfrak{A}_1, \ldots, \mathfrak{A}_r$ are events with $Pr[\mathfrak{A}_2 \cap \cdots \cap \mathfrak{A}_r]\neq 0$, then we have the elementary equality:

$$Pr[\mathfrak{A}_1 \cap \cdots \cap \mathfrak{A}_{r-1} \mid \mathfrak{A}_r] = \prod_{i=1}^{r-1} Pr[\mathfrak{A}_i \mid \mathfrak{A}_{i+1} \cap \cdots \cap \mathfrak{A}_r]. \tag{1}$$

A *stochastic variable* can be any function from $\Omega$ to any arbitrary set. Obviously, a stochastic variable can assume only finitely or enumerably many values. For each value $x$ of the stochastic variable $X$, we put $Pr[X=x]=Pr[X^{-1}(x)]$, where $X^{-1}(x)=\{\omega\in\Omega\colon X(\omega)=x\}$. More generally, if $X_1, \ldots, X_t$ are stochastic variables with values $x_1, \ldots, x_t$, respectively, we put

$$Pr[X_1=x_1, \ldots, X_t=x_t] = Pr[X_1^{-1}(x_1) \cap \cdots \cap X_t^{-1}(x_t)].$$

If confusion is not likely to arise, we shall abbreviate $Pr[X_1=x_1, \ldots, X_t=x_t]$ by $Pr[x_1, \ldots, x_t]$. We say that a stochastic variable $X$ is *uniformly distributed* over a finite set $\Gamma$ if $Pr[X=\gamma]=(\#\Gamma)^{-1}$ for each $\gamma$ in $\Gamma$, where as usual, $\#\Gamma$ denotes the cardinality of $\Gamma$. A stochastic variable $X$ is said to be *independent* of the stochastic variables $Y_1, \ldots, Y_r$ if for all values $x, y_1, \ldots, y_r$ of $X, Y_1, \ldots, Y_r$, respectively, we have $Pr[x,y_1, \ldots, y_r]=Pr[x]Pr[y_1, \ldots, y_r]$.

Let $X\colon\Omega\to\mathfrak{R}_X$ and $Y\colon\Omega\to\mathfrak{R}_Y$ be stochastic variables and let $F\colon\mathfrak{R}_X\to\mathfrak{R}_Y$ be a function. We write $Y=F(X)$ if $Pr[\{\omega\in\Omega : Y(\omega)\neq F(X(\omega))\}]=0$. Obviously, if $Pr[X=x]\neq 0$ then $Pr[Y=y \mid X=x]=1$ if $y=F(x)$ and 0 otherwise.

We denote the set-theoretic difference of the sets $A$ and $B$ by $A \setminus B$. For any set $A$, we denote by $F(A)$ the collection of *finite* subsets of $A$ and by $F^+(A)$ the collection of finite ordered tuples with entries in $A$. Both the empty set and the empty tuple are denoted by $\varnothing$.

## 2.2. Protocols

Informally speaking, a protocol is a description of a *stochastic process* in which *participants*, belonging to a finite *set of participants* $P$, transmit messages between each other which belong to a finite or enumerable *message space* $M$. The time in this stochastic process will be an enumerable set of *moments*, $T=\{0,1,2, \cdots\}$.

The elements of the set $M\times P\times P$ are named *steps*. We shall often denote steps $(m,\alpha,\beta)$ by $\alpha\to\beta : m$ ("$\alpha$ sends $m$ to $\beta$") if $\alpha\neq\beta$, and $\alpha\colon m$ ("$\alpha$ generates $m$") if $\alpha=\beta$.

Let $X=M\times P\times P\times T$. For any subset $y$ of $X$ (including $X$ itself) and subsets $A,B$ of $P$ and $U$ of $T$, we put

$$y(A,B,U) = y \cap (M\times A\times B\times U).$$

Thus $y(A,B,U)$ describes a set of steps in which a participant of $A$ sends a message to a participant in $B$ (or a participant in $A$ generates a message if $A \cap B\neq\varnothing$) during $U$. For convenience

we shall often abbreviate $\{\alpha\}$ by $\alpha$ and $P \setminus \{\alpha\}$ by $P_\alpha$ for $\alpha \in P$, while the subsets $\{t\}$, $\{0, \ldots, t-1\}$, $\{0, \ldots, t\}$, and $\{1,2, \cdots\}$ of $T$ are for convenience written as $t$, $<t$, $\leqslant t$, and $t>0$, respectively.

A stochastic subset of $X$ will be a mapping from $\Omega$ to the collection of subsets of $X$. If $Y$ is such a stochastic subset, then we define $Y(A,B,U)$ by $Y(A,B,U)(\omega) = Y(\omega)(A,B,U)$ for $A,B \subseteq P$ and $U \subseteq T$. Thus if $y$ is a value of $Y$, then $y(A,B,U)$ is the corresponding value of $Y(A,B,U)$.

**Definition 1.** A protocol $\mathcal{P}$ is a tuple $(P,M, p_\alpha : \alpha \in P)$, where $P$ is the (finite) set of participants of $\mathcal{P}$, $M$ is the (finite or enumerable) message space of $\mathcal{P}$ and $p_\alpha$ is the *choice for* $\alpha$.
A choice for $\alpha$ is a collection of functions $\{p_{\alpha,t} : t>0\}$ such that

$$p_{\alpha,t} : X(\alpha,P,<t) \times X(P_\alpha,\alpha,<t) \times F(X(\alpha,P,t)) \to [0,1],$$
$$\sum_{s \in F(X(\alpha,P,t))} p_{\alpha,t}(x,y,s) = 1 \quad \text{for all } x \in X(\alpha,P,<t),\ y \in X(P_\alpha,\alpha,<t). \tag{2}$$

Thus a protocol can be considered as a collection of rules according to which communication between participants takes place. The actual communication is described in the *execution process*:

**Definition 2.** The execution process of the protocol $\mathcal{P} = (P,M,p_\alpha : \alpha \in P)$ is a stochastic subset $S = S_{\mathcal{P}}$ of $X = M \times P \times P \times T$ such that
(i) for every $t \in T$, the values of $S(P,P,t)$ are finite sets;
(ii) $Pr[S(P,P,0) = \varnothing] = 1$;
(iii) for each value $s$ of $S$ and $\alpha \in P$, $t \in T$ we have

$$\begin{aligned} &Pr[s(\alpha,P,t) \mid s(\alpha,P,<t), s(P_\alpha,P,\leqslant t)] \\ &\quad = Pr[s(\alpha,P,t) \mid s(\alpha,P,<t), s(P_\alpha,\alpha,<t)] = p_{\alpha,t}(s(\alpha,P,<t), s(P_\alpha,\alpha,<t), s(\alpha,P,t)), \end{aligned} \tag{3}$$

where $p_\alpha = \{p_{\alpha,t} : t>0\}$ is the choice for $\alpha$.

Values of the execution process are called *executions*. If $s$ is an execution and $(m,\alpha,\beta,t) \in s(\alpha,\beta,t)$ then we say that during execution $s$, $(m,\alpha,\beta)$ is executed by $\alpha$ at moment $t$, and that $m$ is *sent* by $\alpha$ and *received* by $\beta$ at moment $t$ if $\alpha \neq \beta$ and *generated* by $\alpha$ at moment $t$ if $\alpha = \beta$.

Each choice $p_\alpha = \{p_{\alpha,t} : t>0\}$ for $\alpha$ can be considered as a stochastic system (for instance, a mathematical object like a probabilistic Turing machine or a physical object like a computer network) which outputs $s$ at moment $t$ with probability $p_{\alpha,t}(x,y,s)$ after it has been given input $y$ and after it has given output $x$ before moment $t$. The execution process describes the communication between these systems.

(i) states that at each moment, a participant executes only a finite number of steps. (ii) states that at moment 0, "nothing" has happened. (iii) states that whatever a participant does at moment $t$ may depend on all messages which it sent or received before moment $t$, but is not influenced by the messages which it did not send or receive.

It might be possible that for some $\alpha \in P$ and $t \in T$, $S(\alpha,P,t)$ assumes the empty set. In that case $\alpha$ does "nothing" at moment $t$. Protocols with a finite running time, $t_0$, say, can be

considered as protocols for which $Pr[S(P,P,\{t \in T: t > t_0\}) = \varnothing] = 1$.

Our model differs from that of DeMillo, Lynch and Merritt [DLM 82] in that it satisfies the following assumptions:

- each message arrives at the same moment that it is sent, and at the same receiver to which the sender wanted to send its message;
- participant $\gamma$ can find out no more about the communication between $\alpha$ and $\beta$ than what he learns about this from his communication with $\alpha$ and $\beta$; in other words, the communication channel between $\alpha$ and $\beta$ does not "leak";
- the sender and receiver know each others identity.

However, situations in which these assumptions are not valid, can be considered in our model by adding new participants. For instance, weaknesses in a computer network, causing messages to arrive too late or even at the wrong place, or leaking communication channels can be described in our model by considering the computer network or the communication channels as participants of the protocol. (Partial) sender- or recipient-anonymity can be dealt with in our model by giving each participant a number of *representatives*. The representatives communicate with each other and know each other's identities, and each participant communicates with its representatives. Apart from its own representatives, no participant has a priori knowledge about which representatives belong to which participant, and he might find out something about the relationship between the participants and representatives only from the messages which he receives during an execution of the protocol.

From any protocol $\mathcal{P} = (P, M, p_\alpha : \alpha \in P)$ it is possible to construct a new one, by dividing the participants into pairwise disjoint sets, and considering these sets as participants. Let $Q$ be a partition of $P$, i.e. a collection of pairwise disjoint sets of which the union equals $P$. Using (1) and (3) it is possible to show that for each $A$ in $Q$ and each execution $s$ of $\mathcal{P}$,

$$\begin{aligned} &Pr[s(A,P \setminus A,t) \mid s(A,P \setminus A, <t), s(P \setminus A,P, \leqslant t)] \\ &\quad = Pr[s(A,P \setminus A,t) \mid s(A,P \setminus A, <t), s(P \setminus A,A, <t)] \\ &\quad =: p_{A,t}(s(A,P \setminus A, <t), s(P \setminus A,A, <t), s(A,P \setminus A,t)) . \end{aligned} \tag{4}$$

Thus $\mathcal{P}' = \{Q, M, p_A : A \in Q\}$ can be considered as a protocol in which $p_A = \{p_{A,t} : t > 0\}$ is the choice for $A$.

## 2.3. Attacks

In this subsection we consider *attacks* on protocols. If $p_\alpha = \{p_{\alpha,t} : t > 0\}$ and $p'_\alpha = \{p'_{\alpha,t} : t > 0\}$ are two choices for $\alpha$ then $p_\alpha \neq p'_\alpha$ means that for at least one $t$, the functions $p_{\alpha,t}$ and $p'_{\alpha,t}$ are different.

**Definition 3.** Let $\mathcal{P} = (P, M, p_\alpha : \alpha \in P)$ be a protocol and $J$ a (possibly empty) subset of $P$. An attack by $J$ on $\mathcal{P}$ is a protocol $\mathcal{P}' = (P, M, p'_\alpha : \alpha \in P)$ such that

$p_\alpha \neq p'_\alpha$ for $\alpha \in J$ , $p_\alpha = p'_\alpha$ for $\alpha \in P \setminus J$ .

We say that the participants in $J$ are *cheating*.

Thus an attack can be interpreted as a violation of the rules of a protocol by some of the participants.

By considering computer networks or communication channels as participants, it is possible to describe attacks such as passive or active eavesdropping, or redirection of messages. By using representatives, as introduced in the previous subsection, our model allows attacks to be described in which some participant pretends to be somebody else.

In the security analysis of protocols, it is important to know whether non-cheating participants are able to find out if other participants are cheating. Below, a precise definition of *detection* of an attack is given.

**Definition 4.** Let $\mathcal{P} = (P, M, p_\alpha : \alpha \in P)$ be a protocol, $J$ a subset of $P$ and $\mathcal{P}'$ an attack by $J$ on $\mathcal{P}$. Denote by $S_\mathcal{P}, S_{\mathcal{P}'}$ the execution processes of $\mathcal{P}$ and $\mathcal{P}'$, respectively, and let $s$ be an execution of $\mathcal{P}'$. We say that $\alpha \in P \setminus J$ can *detect* $\mathcal{P}'$ during $s$ if

$$Pr[S_\mathcal{P}(P_\alpha, \alpha, T) = s(P_\alpha, \alpha, T)] = 0 ,$$

whereas

$$Pr[S_{\mathcal{P}'}(P_\alpha, \alpha, T) = s(P_\alpha, \alpha, T)] > 0 .$$

One possible way by which $\alpha$ may detect an attack on the protocol $\mathcal{P}$ is when at some moment $t$ he receives messages from a participant $\beta$ which are not *allowed* for $\mathcal{P}$. By this we mean that, given the communication between $\alpha$ and $\beta$ before moment $t$, $\alpha$ received messages from $\beta$ at moment $t$ which he could not have received with positive probability during an execution of $\mathcal{P}$. (This need not imply that $\beta$ is cheating). We now express this by means of the terminology introduced above. Let $S_\mathcal{P}$ denote the execution process of $\mathcal{P}$, and let $s$ be an execution of (an attack on) $\mathcal{P}$. Thus $s(\alpha, \beta, <t)$ and $s(\beta, \alpha, <t)$ describe the communication between $\alpha$ and $\beta$ before moment $t$, during $s$. Then the messages sent from $\beta$ to $\alpha$ at moment $t$ during execution $s$ are *allowed* for $\mathcal{P}$ if

$$Pr[S_\mathcal{P}(\beta, \alpha, t) = s(\beta, \alpha, t) \mid S_\mathcal{P}(\beta, \alpha, <t) = s(\beta, \alpha, <t) , \; S_\mathcal{P}(\alpha, \beta, <t) = s(\alpha, \beta, <t)] > 0 . \qquad (5)$$

In the situation that we are dealing with cryptographic protocols, participants often have limited computational abilities and therefore limited possibilities to cheat. To incorporate this in our model we assume that each participant $\alpha$ of some protocol $\mathcal{P}$ has a collection of choices $\mathcal{C}_\alpha$, each element of which satisfies (2). Then each attack $\mathcal{P}' = (P, M, p'_\alpha : \alpha \in P)$ on $\mathcal{P}$ must satisfy

$$p'_\alpha \in \mathcal{C}_\alpha \text{ for } \alpha \in P . \qquad (6)$$

## 3. DESCRIPTION OF THE CREDENTIAL MECHANISM

In §3.1 we explain the main idea behind the credential mechanism. §3.2 contains a more detailed overview of the credential mechanism. In §3.3 we give a concise description of the credential mechanism by means of a convenient protocol language. This description of the credential mechanism will be referred to throughout the paper. §3.4 contains additional comments on the credential mechanism. §§3.1-3.4 can be read independently of §2. Finally, in §3.5 we describe a mathematical model for the credential mechanism by means of the formalism introduced in §2. There, all the notions introduced in §§3.2-3.4 will be given a precise mathematical meaning.

### 3.1. Main idea behind the credential mechanism

Our credential mechanism is a cryptographic protocol based on RSA used with a single composite modulus $N$. The participants of the credential mechanism are individuals and organizations. $N$ is public, i.e. known to all participants of the credential mechanism; only one special organization in the credential mechanism, the "signature authority" $Z$, knows how to factor $N$. The messages transmitted in the credential mechanism belong to $\mathbf{Z}_N^*$, which is the multiplicative group of all residue classes modulo $N$ containing integers coprime with $N$. The order of $\mathbf{Z}_N^*$ is as usual denoted by $\phi(N)$. Only $Z$ has the ability to compute RSA-signatures on these messages. An RSA-signature on message $m$ is a message $m^{\bar{c}} \bmod N$, where $c$ is a public integer coprime with $\phi(N)$, and $\bar{c}$ is an integer with $c\bar{c} \equiv 1 \bmod \phi(N)$, which is known only to $Z$. The credentials will be public positive integers coprime with $\phi(N)$, belonging to a finite set $C$. The product of all elements of $C$ is denoted by $b$.

Suppose $i$ is an individual participating in the credential mechanism. The pseudonyms used by $i$ are formed as follows: first $i$ gets a number $u$ from $Z$ which $i$ uses as a pseudonym with $Z$; then $i$ generates, for each organization $A$ participating in the credential mechanism, a random number $r_A$ from $\mathbf{Z}_N^*$. Then $i$ uses as pseudonym with $A$ the number $u_A \equiv u r_A^b \bmod N$, where $b$ is the product of all credentials. For the organizations, these pseudonyms just look like random numbers in $\mathbf{Z}_N^*$; this prevents different organizations from linking the pseudonyms used by the same individual.

A credential $c \in C$, applying to individual $i$, can be sent from organization $A$ to organization $B$ as follows:

- $A$ asks $Z$ to compute $d_A \equiv u_A^{\bar{c}} \bmod N$ for him. After $A$ receives this he sends $d_A$ to $i$. $i$ checks if $A$ sent him the correct message by verifying that $d_A^c \equiv u_A \bmod N$.
- $i$ computes $d_B \equiv u_B^{\bar{c}}$ by first dividing $d_A$ by $r_A^{b/c}$ and then multiplying with $r_B^{b/c}$. (Note that the exponent $b / c$ is the product of all credentials except $c$ so that $i$ can compute it).
- $i$ sends $d_B$ to $B$ and $B$ verifies that $d_B^c \equiv u_B \bmod N$.

Individuals should never be able to show a credential to some organization if they did not

get this credential before from another organization. Individuals might be able to compute credentials themselves, without having gotten them from some organization, if they have the freedom to construct their pseudonyms $u_A$ in another way than described above. If for instance $i$ can use $u_B \equiv r_B^b \bmod N$ as a pseudonym with $B$ instead of $ur_B^b \bmod N$, then he can compute each credential $u_B^{\bar{c}} \bmod N$ by himself. Moreover, if two individuals $i$ and $i'$ use pseudonyms $u_B$ and $u_B' \equiv u_B r^{b_B} \bmod N$ with $B$, where $r$ is chosen by both individuals, then maybe $i$ needs to get a credential from some other organization before he can compute one for $B$; but once $i$ is able to show a credential to $B$, $i'$ is able to show the same credential to $B$, without having gotten it from some other organization.

To avoid the problems just mentioned, we extended our credential mechanism with a validating part, which forces individuals to form their pseudonyms $u_A$ in the way described above, but does not require individuals to reveal more about how they have actually constructed their pseudonyms. In the validating part for pseudonym $u_A$, $i$ sends messages to $Z$, which are constructed in a special way, by means of a one-way function. These messages are *candidates* for building blocks of a *validator*, to be issued by $Z$ to $i$ later on. Then $Z$ selects at random half of these candidates, and asks $i$ to show how he actually constructed these. If $i$ constructed these properly, then $Z$ computes the validator from the candidates of which $i$ did not reveal the construction, and submits this validator to $i$. Because there is an RSA-signature in the validator, $i$ could not have computed the validator by himself. Later, $i$ transforms this validator into another validator which is shown to $A$ together with pseudonym $u_A$. There must be a special relationship between $u_A$ and this validator which is checked by $A$. If this relationship holds, $A$ accepts $u_A$ as a pseudonym. $Z$ also checks this relationship, to make sure that later he does not issue credentials on improperly formed pseudonyms.

### 3.2. Overview of the credential mechanism

In the actual credential mechanism, it does not make a difference whether some individual communicates with some organization, thereby identifying himself with a pseudonym, or some *representative* of this individual communicates with that organization and identifies himself with that pseudonym. In our description of the credential mechanism, we shall assume that communication takes place between organizations and representatives. Thus the participants in the credential mechanism will be the signature authority $Z$, the organizations $A_1, \ldots, A_L$, the individuals $i_1, \ldots, i_R$ and the individuals' representatives, where no representative represents more than one individual and each individual has different representatives for the communication with different organizations.

**Initialization.** The notation introduced here will be used throughout the remainder of this paper and will not be re-introduced later. Before the actual credential mechanism starts, $Z$ chooses two large primes $P$ and $Q$ and keeps these secret. Then $Z$ makes the modulus $N = PQ$ public. After that, $Z$ makes public:
a set $C = \{c_1, \ldots, c_K\}$ of positive integers, to be used as credentials;

pairs of primes $(p_1,q_1), \ldots, (p_L,q_L)$ used to make validators for $A_1, \ldots, A_L$, respectively;
an even integer $n>4$, the *security parameter*, which determines the amount of work done in the validating part;
a positive integer $a$, elements $m_1, \ldots, m_n$ of $\mathbf{Z}_N^*$ and a "one-way function" $f:\mathbf{Z}_N^* \to \mathbf{Z}_N^*$, which are all used in the validating part.

It is assumed that the numbers $\phi(N)$, $a, p_1, \ldots, p_L, q_1, \ldots, q_L, c_1, \ldots, c_K$ are pairwise coprime, and that $p_1, \ldots, p_L, q_1, \ldots, q_L$ are larger than $\frac{1}{2}n$. (This last condition is just a technical one, needed later in some of our arguments). We put $b_j = p_j^2 c_1 \cdots c_K$ for $j = 1, \ldots, L$. We shall not specify $f$. We assume that $f$ is "close to random" and has at least the properties that anybody can easily compute it and that it is not a homomorphism and not invertible by any participant of the credential mechanism (possibly apart from $Z$). (One could take $f(x) = x^{\pi_1} + x^{\pi_2} + 1 \bmod N$ for certain prime numbers $\pi_1$ and $\pi_2$ or another polynomial which is not a power of a linear polynomial. But we do not know if such a choice for $f$ is good enough). All computations, relations etc. in the credential mechanism will be modulo $N$, and therefore the suffix "mod $N$" will be omitted. If $b$ is a public number, used for exponentiation in RSA then the corresponding secret exponent, known only to $Z$, is denoted by $\bar{b}$. Thus $b\bar{b} \equiv 1 \bmod \phi(N)$. In general, all expressions in exponents without a bar will be integers computable by all participants of the credential mechanism, while only $Z$ is able to compute exponents with a bar.

Below we give an overview of the whole credential mechanism. It is built up from *subprotocols* (i.e. collections of steps to be executed in the credential mechanism) which are indicated by roman digits and the individual or representative, organization and credential involved. We describe that part of the credential mechanism in which $i_k$ is involved. The representative of $i_k$ communicating with $A_j$ is denoted by $g_j$.

In most of the subprotocols, some participant checks if the messages which it received satisfy some special relationship. The check results in 'true' if this relationship holds, and 'false' otherwise. No steps in a subprotocol are executed after one of its checks has resulted in 'false'.

***O*($i_k$)**: $i_k$ gets a pseudonym from $Z$.
- $i_k$ asks $Z$ for a pseudonym.
- $Z$ checks if a pseudonym can be issued to $i_k$, and if this is the case, chooses $u_k$ from $\mathbf{Z}_N^*$ and sends this as a pseudonym to $i_k$.

***I*($i_k$,$A_j$)**: $i_k$ gets a validator from $Z$ which will be shown to $A_j$ in a modified form. Put $p = p_j, q = q_j, b = b_j$.
- $i_k$ asks $Z$ for a validator for $A_j$ and $Z$ decides if this can be issued.
- $i_k$ chooses numbers $r_l, s_l$ $(l = 1, \ldots, n)$ at random from $\mathbf{Z}_N^*$. Then he computes $\tilde{r}_l := m_l r_l^q$ and $a_{kl} := f(u_k \tilde{r}_l^b) s_l^{pq}$ for $l = 1, \ldots, n$ and sends all $a_{kl}$ to $Z$. (A uniform choice from $\mathbf{Z}_N^*$ can be obtained by choosing an integer uniformly from $\{1, \ldots, N\}$, by doing another choice in the unlikely event that this integer has a factor in common with $N$ and so on, until an integer coprime with $N$ is chosen).

The numbers $a_{kl}$ are the candidates for the building blocks of the validator which will be issued

by $Z$ later on. These candidates are constructed in a special way but by the uniform choice of the numbers $r_l, s_l$ they look just like random elements of $\mathbf{Z}_N^*$. This special way of construction should give the organizations sufficient security; while the random choice of the numbers $r_l, s_l$ is meant for giving individuals the desired privacy.

- $Z$ selects at random a subset $\mathbb{S}$ of $\{1, \ldots, n\}$ of cardinality $\frac{1}{2}n$ and asks $i_k$ to show, for each $l$ in $\mathbb{S}$, numbers $r_l$ and $s_l$ such that $a_{kl} = f(u_k(m_l r_l^a)^b) s_l^{pq}$. Then $Z$ checks if $a_{kl}$ and the numbers $r_l, s_l$ sent to it by $i_k$ satisfy these relationships for $l \in \mathbb{S}$.
- If these relationships are satisfied, then $Z$ computes the validator

$$v_{kj} := u_k^{\overline{p}^2} \{ \prod_{l \notin \mathbb{S}} a_{kl} \}^{\overline{pq}}$$

and sends this to $i_k$. ($\overline{p}^2$ denotes a secret exponent corresponding to $p^2$). Then $i_k$ checks if this validator has the proper form by verifying that

$$v_{kj}^{p^2 q} = u_k^q \{ \prod_{l \notin \mathbb{S}} a_{kl} \}^p .$$

***II*($g_j, A_j$).** $i_k$ forms a pseudonym $u_{g_j}$ and a validator $\tilde{v}_{g_j}$ from $u_k$ and $v_{kj}$, and lets his representative $g_j$ show these to $A_j$. Let again $p = p_j, q = q_j, b = b_j$, and put $t_1 = 1$.

- Let $\sigma$ be a permutation of $\{1, \ldots, n\}$ with $\sigma(\{\frac{1}{2}n + 1, \ldots, n\}) = \mathbb{S}$, where $\mathbb{S}$ is the set chosen by $Z$ in $I(i_k, A_j)$. Let $r'_l = \tilde{r}_{\sigma(l)}$, $s'_l = s_{\sigma(l)}$ for $l = 1, \ldots, \frac{1}{2}n$.

Put $r_{g_j} := r'_1$. $i_k$ computes $u_{g_j} := u r_{g_j}^b$, which will be the pseudonym used with $A_j$, and $w_{g_j} = \prod_{l=2}^{\frac{1}{2}n} u_k r'^b_l$. Then $i_k$ sends $u_{g_j}$ and $w_{g_j}$ to $g_j$, $g_j$ sends these to $A_j$ and finally, $A_j$ sends these numbers to $Z$.

- $A_j$ and $Z$ check if $u_{g_j} \neq u_{g'_j}$ and $u_{g_j} w_{g_j} \neq u_{g'_j} w_{g'_j}$ for each pair $u_{g'_j}, w_{g'_j}$ sent to $A_j$ by some other representative $g'_j$.

Each individual $i_k$ forms a validator to be shown to $A_j$ from the validator issued by $Z$ in $I(i_k, A_j)$. The last check prevents different individuals from computing their validators for $A_j$ from the same validator issued by $Z$.

- $i_k$ computes $t_l := r'_l r_{g_j}^{-1}$ for $l = 2, \ldots, \frac{1}{2}n$, and

$$\tilde{v}_{g_j} = r_{g_j}^{b/p^2} (s'_1 \cdots s'_{\frac{1}{2}n})^{-1} \times v_{kj}$$

and sends these numbers to $g_j$. Then $g_j$ shows these numbers to $A_j$ and $A_j$ sends them to $Z$.

A straightforward computation shows that

$$\tilde{v}_{g_j} = (u r_{g_j}^b)^{\overline{p}^2} \left\{ \prod_{l=1}^{\frac{1}{2}n} f(u r_{g_j}^b t_l^b) s'^{pq}_l \right\}^{\overline{pq}} (s'_1 \cdots s'_{\frac{1}{2}n})^{-1} = u_{g_j}^{\overline{p}^2} \{ \prod_{l=1}^{\frac{1}{2}n} f(u_{g_j} t_l^b) \}^{\overline{pq}} .$$

- Both $A_j$ and $Z$ check if

$w_{g_j} = \prod_{l=2}^{½n} u_{g_j} t_l^{pq}$ , and $\tilde{v}_{g_j}^{p^2 q} = u_{g_j}^{q} \{ \prod_{l=1}^{½n} f(u_{g_j} t_l^{b}) \}^{p}$ .

If these conditions hold then both $A_j$ and $Z$ accept $u_{g_j}$.

***III*($g_j$,$A_j$,$c$)**: $A_j$ issues credential $c$ on the pseudonym of representative $g_j$.

- $g_j$ asks $A_j$ for credential $c$ on his pseudonym at the request of $i_k$. Then $A_j$ checks if this can be issued and, after having positively decided on this, asks $Z$ to compute credential $c$ on the pseudonym $u_{g_j}$ of $g_j$.
- $Z$ checks if $u_{g_j}$ has been shown with a proper validator, and if so, computes $d_{g_j} := u_{g_j}^{\bar{c}}$ and sends this credential to $A_j$. $A_j$ issues this credential to $g_j$ and $g_j$ gives it to $i_k$.
- $i_k$ checks if $d_{g_j}^{c} = u_{g_j}$, and computes $d_k := d_{g_j} r_{g_j}^{-b_j / c} = u_k^{\bar{c}}$.

***IV*($g_h$,$A_h$,$c$)**: $g_h$ shows credential $c$ on his pseudonym with $A_h$.

- $i_k$ computes $d_{g_h} = d_k r_{g_h}^{b_h / c} = u_{g_h}^{\bar{c}}$ and sends this to $g_h$. $g_h$ shows this credential to $A_h$, and $A_h$ checks if $d_{g_h}^{c} = u_{g_h}$.

## 3.3. Concise description of the credential mechanism.

We shall use the notation of §3.2. Further notation introduced here will be used later without re-introduction. For reference purposes we describe the credential mechanism by means of a "protocol language" introduced below.

**Protocol language.**

$\alpha \rightarrow \beta$ : $m$

> $\alpha$ sends message $m$ to $\beta$

$\alpha \rightarrow \beta \rightarrow \gamma \rightarrow \cdots \rightarrow \kappa$ : $m$

> $\alpha$ sends $m$ to $\beta$, $\beta$ sends $m$ to $\gamma$ $\cdots$ to $\kappa$

$\alpha$ : chooses $\gamma$ from $\Gamma$

> $\alpha$ chooses an element $\gamma$ from the set $\Gamma$ in some unspecified way

$\alpha$ : chooses $\gamma$ uniformly from $\Gamma$

> $\alpha$ chooses $\gamma$ uniformly from the set $\Gamma$ and independently of all other steps executed at the same moment or before

$\alpha$ : computes $m := (expression)$

> $\alpha$ computes the expression and assigns the value to $m$

$\alpha \rightarrow \beta \rightarrow \cdots \rightarrow \kappa$ : checks $P$

> $\alpha$ computes the value of the predicate $P$, which is either 'true' or 'false'. Then this value is sent from $\alpha$ to $\beta$, from $\beta$ to $\cdots$ to $\kappa$. If this value is 'true' the subprotocol continues; otherwise the subprotocol stops immediately after the check.

**Description of the credential mechanism.** We shall describe that part of the credential mechanism in which individual $i_k$ is involved. The representative of $i_k$ communicating with $A_j$ is

denoted by $g_j$, for $j = 1, \ldots, L$. First we describe the initialization, before the actual credential mechanism starts.

**Initialization**

$Z$ : chooses large primes $P,Q$
  computes $N := PQ$
  chooses even integer $n > 4$ and one-way function $f$
  chooses $m_1, \ldots, m_n$ from $\mathbf{Z}_N^*$
  chooses positive integers $c_1, \ldots, c_K, a, p_1, \ldots, p_L, q_1, \ldots, q_L$,
    such that all these numbers are pairwise coprime and coprime with $\phi(N)$
    and $p_1, \ldots, p_L, q_1, \ldots, q_L$ are primes larger than $\frac{1}{2}n$
  computes $\bar{c}_1, \ldots, \bar{c}_K, \bar{a}, \bar{p}_1, \ldots, \bar{p}_L, \bar{q}_1, \ldots, \bar{q}_L$
$Z$ makes public: $N, n, f, m_1, \ldots, m_n, c_1, \ldots, c_K, a, p_1, \ldots, p_L, q_1, \ldots, q_L$

***O*($i_k$).** $Z$ gives pseudonym to $i_k$.

1. $i_k \to Z$ : asks for pseudonym
2. $Z \to i_k$ : checks if pseudonym can be given
3. $Z$ : chooses $u_k$ from $\mathbf{Z}_N^*$
4. $Z \to i_k$ : $u_k$

***I*($i_k$,$A_j$).** $Z$ gives $i_k$ validator for $A_j$.
Put $p = p_j, q = q_j, b = b_j$.

1. $i_k \to Z$ : asks for validator for $A_j$
2. $Z \to i_k$ : checks if a validator for $A_j$ can be issued
3. $i_k$ : chooses $(r_l, s_l : l = 1, \ldots, n)$ uniformly from $(\mathbf{Z}_N^*)^{2n}$
4. $i_k$ : computes $\tilde{r}_l := m_l r_l^a$, $a_{kl} := f(u_k \tilde{r}_l^{\bar{b}}) s_l^{pq}$ $(l = 1, \ldots, n)$
5. $i_k \to Z$ : $a_{kl}$ $(l = 1, \ldots, n)$
6. $Z$ : chooses $\mathbb{S}$ uniformly from $\{\mathbb{S} \subset \{1, \ldots, n\} : \#\mathbb{S} = \frac{1}{2}n\}$
7. $Z \to i_k$ : $\mathbb{S}$
8. $i_k \to Z$ : $r_l, s_l$ $(l \in \mathbb{S})$
9. $Z \to i_k$ : checks $a_{kl} = f(u_k (m_l r_l^a)^b) s_l^{pq}$ for $l \in \mathbb{S}$
10. $Z$ : computes $v_{kj} := u_k^{\bar{p}^2} (\prod_{l \notin \mathbb{S}} a_{kl})^{\bar{p}\bar{q}}$
11. $Z \to i_k$ : $v_{kj}$
12. $i_k \to Z$ : checks $v_{kj}^{p^2 q} = u_k^q (\prod_{l \notin \mathbb{S}} a_{kl})^p$

***II*($g_j$,$A_j$).** $g_j$ shows pseudonym with validator to $A_j$.
Let $r_l, s_l$ be the numbers chosen in step 3 of $I(i_k, A_j)$, let $\mathbb{S}$ the set chosen by $Z$ in step 6, and let $\sigma$ be the permutation of $(1, \ldots, n)$ defined by $\sigma(1) < \sigma(2) < \cdots < \sigma(\frac{1}{2}n)$, $\sigma(\frac{1}{2}n + 1) < \cdots < \sigma(n)$ and $\sigma(\{\frac{1}{2}n + 1, \ldots, n\}) = \mathbb{S}$. Put $p = p_j, q = q_j, b = b_j$ and $t_1 = 1$.

1. $i_k$ : computes $r_{g_j} := \tilde{r}_{\sigma(1)}$, $u_{g_j} := u_k r_{g_j}^b$, $w_{g_j} := \prod_{l=2}^{\frac{1}{2}n} u_k \tilde{r}_{\sigma(l)}^{\bar{b}}$

2. $i_k \to g_j \to A_j \to Z$ : $u_{g_j}, w_{g_j}$
3. $Z \to A_j \to g_j \to i_k$ : checks $u_{g_j} \neq u_{g'_j}$ and $u_{g_j} w_{g_j} \neq u_{g'_j} w_{g'_j}$ for any pair $u_{g'_j}$, $w_{g'_j}$ submitted to $A_j$ by another representative $g'_j$.
4. $i_k$ : computes $t_l := \dfrac{\tilde{r}_{\sigma(l)}}{r_{g_j}}$ $(l = 2, \ldots, \frac{1}{2}n)$,
$$\tilde{v}_{g_j} := r_{g_j}^{b/p^2} \left(\prod_{l=1}^{\frac{1}{2}n} s_{\sigma(l)}\right)^{-1} v_{kj} \qquad \left(= u_{g_j}^{\bar{p}^2} \left(\prod_{l=1}^{\frac{1}{2}n} f(u_{g_j} t_l^b)\right)^{\bar{p}\bar{q}}\right)$$
5. $i_k \to g_j \to A_j \to Z$ : $t_l$ $(l = 2, \ldots, \frac{1}{2}n)$, $\tilde{v}_{g_j}$
6. $Z \to A_j \to g_j \to i_k$ : checks $\tilde{v}_{g_j}^{p^2 q} = u_{g_j}^q \left(\prod_{l=1}^{\frac{1}{2}n} f(u_{g_j} t_l^b)\right)^p$, $w_{g_j} = \prod_{l=2}^{\frac{1}{2}n} u_{g_j} t_l^b$

***III*($g_j$,$A_j$,$c$).** $A_j$ issues credential $c$ to $g_j$.
Let $r_{g_j}, u_{g_j}$ have the same meaning as in step 1 of $II(g_j, A_j)$.

1. $i_k \to g_j \to A_j$ : asks for credential $c$
2. $A_j \to g_j \to i_k$ : checks if $c$ can be issued
3. $A_j \to Z$ : $u_{g_j}$
4. $Z \to A_j$ : checks if it has received proper validator for $u_{g_j}$
5. $Z$ : computes $d_{g_j} := u_{g_j}^{\bar{c}}$
6. $Z \to A_j \to g_j \to i_k$ : $d_{g_j}$
7. $i_k \to g_j \to A_j \to Z$ : checks $d_{g_j}^c = u_{g_j}$
8. $i_k$ : computes $d_k := r_{g_j}^{-b_j/c} d_{g_j}$ $(= u_k^{\bar{c}})$

***IV*($g_h$,$A_h$,$c$).** $g_h$ shows credential $c$ to $A_h$.
$d_k$ will be the number computed in step 8 of $III(g_j, A_j, c)$ and $u_{g_h}, r_{g_h}$ will be the numbers computed in step 1 of $II(g_h, A_h)$.

1. $i_k$ : computes $d_{g_h} := d_k r_{g_h}^{b_h/c}$ $(= u_{g_h}^{\bar{c}})$
2. $i_k \to g_h \to A_h$ : $d_{g_h}$
3. $A_h \to g_h \to i_k$ : checks $d_{g_h}^c = u_{g_h}$

### 3.4. Comments

This subsection contains some remarks and assumptions on the credential mechanism.

**Outside world.** Participants of the credential mechanism may not only communicate with each other, mostly over some computer network, but also with the "outside world". For instance, events happening in the outside world may influence an organization's decision to issue a credential, or an individual's decision to ask for or show a credential. For that reason, the outside world should be considered as another participant of the credential mechanism.

**Time order of the steps.** The credential mechanism is built up from the subprotocols in the set $\Pi$ consisting of

$O(i_k), I(i_k,A_j), II(g_j,A_j), III(g_j,A_j,c), IV(g_j,A_j,c)$

for all individuals $i_k$, representatives $g_j$ and organizations $A_j$. In each subprotocol $\mathcal{P}$ of $\Pi$, certain relationships between messages are checked, resulting in the value 'true' if the relationship holds and 'false' otherwise. We agreed that no steps in $\mathcal{P}$ are executed after one of the checks in $\mathcal{P}$ resulted in 'false'. We say that a subprotocol in $\Pi$ has been *properly executed* if it has been executed such that none of its checks resulted in 'false'. We require that subprotocols are executed without any interruption. We allow that different subprotocols in $\Pi$ run in parallel or in overlapping time intervals, however with the following obvious

**Consistency restriction:**

- steps in $I(i_k,A_j)$ are executed only after $O(i_k)$ has been properly executed;
- if $g_j$ is the representative of $i_k$ communicating with $A_j$, then steps in $II(g_j,A_j)$ are executed only after $I(i_k,A_j)$ has been properly executed;
- steps in $III(g_j,A_j,c)$ are executed only after $II(g_j,A_j)$ has been properly executed;
- steps in $IV(g_h,A_h,c)$ are executed only after $III(g_j,A_j,c)$ has been properly executed for some $j$ in $\{1, \ldots, L\}$ and some $g_j$ representing the same individual as $g_h$.

There might be more restrictions on the time order in which the steps in the credential mechanism are executed, for instance:

- pseudonyms, validators or credentials must be issued before some "deadline";
- the time passing between the moment that an individual or its representative gets a pseudonym, validator or credential from an organization, and the moment that another representative of this individual shows this to another organization, depends statistically on events in the outside world;
- the decision of an organization about issuing a particular credential on a pseudonym depends on whether other credentials have been shown on that pseudonym, on the number of times that that credential has been issued before on other pseudonyms, or on messages received from the outside world.

**Simple credential mechanism.** A possible way to state properties of the credential mechanism, is to compare it with the following simple credential mechanism.

When some organization agrees to give a credential $c$ to an individual, that organization just gives the individual's representative the number $c$, without any cryptographic protection. Later on, another representative of that individual shows this number $c$ to another organization. The validating part of this credential mechanism runs as follows: individual $i_k$ asks $Z$ for permission to communicate with organization $A_j$. If $Z$ gives this permission then he sends a special validator $v_j$ to $i_k$ which is independent of $i_k$. Later, $i_k$ initiates the conversation with $A_j$ by letting his representative show $v_j$ to $A_j$.

Obviously, this simple credential mechanism does not give the organizations any security;

individuals are always able to show a validator or credential to some organization by means of their representatives without having got this from another organization. This simple credential mechanism would work well if none of the individuals would ever cheat; compared with other possible credential mechanisms, it gives individuals maximal possibilities of getting validators or credentials and maximal freedom in choosing the moments at which they show these to some organization.

**Main condition:** except for cheating, our credential mechanism should give individuals as many abilities as the simple credential mechanism described above, more precisely:

the credential mechanism must offer each participant the same freedom in communicating with the outside world, and each non-cheating individual the same possibilities of getting validators or credentials and the same freedom in deciding when to ask for or show these to some organization, as the simple credential mechanism described above.

**Checks.** The checks done in the credential mechanism are divided in two parts: the *decision checks* in which organizations check if they can issue a pseudonym, validator or credential to some individual or representative; and the other, so-called *security checks* by which participants may detect attacks. In any execution of the credential mechanism, no step in a subprotocol $\mathcal{P}$ in $\Pi$ is executed after some check in $\mathcal{P}$ has resulted in 'false'; we allow however that the execution of a subprotocol is repeated after a security check by an individual has resulted in 'false'. If no participant cheats, then all security checks will give the value 'true'; only the checks in steps 3 of the sets $\Pi(g_j,A_j)$ may give the value 'false' with very small probability.

We now briefly discuss the checks in steps 3 of the sets $\Pi(g_j,A_j)$. These checks differ from the other security checks in that they compare messages which were sent by different representatives to an organization. Without these checks, two individuals, $i_1$ and $i_2$, say, can successfully conspire in the following way against $A_j$: $i_1$ follows the validating part and lets his representative $g_{1j}$ show $u_1,w_1$ to $A_j$ in step 2, and $v_1,t_{12},\ldots,t_{1,½n}$ in step 5 of $\Pi(g_{1j},A_j)$, where

$$w_1=\prod_{l=2}^{½n} u_1 t_{1l}^b\ ,\ v_1=u_1^{\bar{p}^2}\ \{f(u_1)\prod_{l=2}^{½n} f(u_1 t_{1l}^b)\}^{\overline{pq}}\ .$$

($b=b_j$, $p=p_j$ and $q=q_j$ have the same meaning as in §3.3). $i_2$ chooses $u_2=u_1 t_{1m}^b$ for some $m$ in $\{1,\ldots,½n\}$, where $t_{11}:=1$, and computes $w_2:=u_1 w_1 / u_2$, $v_2:=t_{1m}^{b/p^2} v_1$ and $t_{2l}:=t_{1,\pi(l)} / t_{1m}$ for $l=2,\ldots,½n$, where $\tau$ is a permutation of $\{1,\ldots,½n\}$ with $\tau(m)=1$. Then $i_2$ lets his representative $g_{2j}$ show $u_2$, $w_2$, $v_2$ and $t_{2l}$ ($l=2,\ldots,½n$) to $A_j$ in $\Pi(g_{2j},A_j)$. All security checks by $A_j$ in $\Pi(g_{2j},A_j)$ are satisfied, except that $u_1w_1=u_2w_2$, hence without the check in step 3, $A_j$ would have accepted both $u_1$ and $u_2$. It is easy to check that credentials issued on $u_1$ can be easily transformed into credentials on $u_2$ and vice-versa.

Suppose that $A_j$ received $u_1$ and $w_1$ from representative $g_{1j}$ in step 2 of $\Pi(g_{1j},A_j)$. Problems might arise when there is a dispute in which $A_j$ claims that it received numbers $u_2$ and $w_2$ from another representative $g_{2j}$ such that $u_1=u_2$ or $u_1w_1=u_2w_2$ and refuses to accept pseudonym $u_1$, and that $g_{1j}$ does not accept this refusal. Below we describe a method to deal with

such a dispute with the help of a mutually trusted referee, in such a way that $g_{1j}$ does not have to reveal which individual it represents. We assume that $\frac{1}{2}n$, where $n$ is the security parameter, is coprime with all the primes $p_j, q_j$ and credentials $c$ introduced in §3.2.

For $i=1,2$, let $u_i$ and $w_i$ be the numbers sent to $A_j$ in step 2, $v_i = \bar{v}_{g_{ij}}$ the validator computed in step 4 and $\tau_i$ the tuple $(t_{i2}, \ldots, t_{i,\frac{1}{2}n})$ sent to $A_j$ in step 5, where all steps belong to $\Pi(g_{ij}, A_j)$. In the case of a dispute as described above, each $g_{ij}$ sends $u_i$, $w_i$, $v_i$ and $\tau_i$ to the referee. First the referee computes the tuples $\sigma_i = (u_i, u_i t_{il}^b : l=2, \ldots, \frac{1}{2}n)$ for $i=1,2$. Note that none of these tuples needs contain distinct entries. Then the referee checks, if indeed $u_1 = u_2$ or $u_1 w_1 = u_2 w_2$ and

$$v_i^{p^2 q} = u_i^q \left[ \prod_{l=1}^{\frac{1}{2}n} f(u_i t_{il}^b) \right]^p, \quad w_i = \prod_{l=2}^{\frac{1}{2}n} u_i t_{il}^b \quad \text{for } i=1,2. \tag{7}$$

If (7) holds and the two tuples $\sigma_1$ and $\sigma_2$ can be made equal by reordering, then the referee concludes that the individual represented by $g_{1j}$ conspired with somebody else and decides that $A_j$ does not have to accept $u_1$ as a pseudonym. His motivation for this conclusion is the following: since for any integer $d$ with $\gcd(d, \phi(N)) = 1$ the mapping $x \mapsto x^d$ is bijective, the tuple $\sigma_1$ can be considered as a random tuple in $(\mathbf{Z}_N^*)^{\frac{1}{2}n}$. The number of tuples $\sigma_1$ in $(\mathbf{Z}_N^*)^{\frac{1}{2}n}$ which contain $u_1$ and whose other $\frac{1}{2}n-1$ entries have product $w_1$ is equal to $\phi(N)^{\frac{1}{2}n-2}$. If $g_{1j}$, or the individual which it represents, did not reveal the set $\sigma_1$ before showing it to the referee, then somebody else could have generated $u_2$ and $\tau_2$ such that $\sigma_1$ equals $\sigma_2$ after reordering, only by correctly guessing a tuple which is apart from its order equal to $\sigma_1$, while knowing no more than $u_1$ and $w_1$. But the chance of such a correct guess is at most $(\frac{1}{2}n)! \times \phi(N)^{2-\frac{1}{2}n}$. In practical situations when $N$ has about 200 digits, this probability can be neglected.

If $\sigma_1$ can not be made equal to $\sigma_2$ by reordering, then the referee accuses signature authority $Z$ of cheating. From $u_1 w_1 = u_2 w_2$ it follows that by cooperation, $g_{1j}$ and $g_{2j}$ can compute $(u_1 u_2^{-1})^{\bar{b}}$. (Of course, $g_{1j}$ and $g_{2j}$ can compute this also if $u_1 = u_2$). The referee assumes that participants in the credential mechanism other than $Z$ have only a negligibly small chance of learning at the same time $u_1, r \in \mathbf{Z}_N^*$ and $u_2 := u_1 r^b$, tuples $\tau_i = (t_{i2}, \ldots, t_{i,\frac{1}{2}n})$ of $\mathbf{Z}_N^*$ and validators $v_i$ satisfying (7) for $i=1,2$, such that no reordering of $\sigma_1$ is equal to $\sigma_2$. Theorem 3 in §7.2 can be considered as a motivation for this.

### 3.5. A mathematical model for the credential mechanism

In this subsection we shall describe the credential mechanism by means of the terminology introduced in §2. To this end, we must interpret each step described in §3.3 as a step in an execution of a protocol in the sense of §2, and give the notions introduced in §3.4 a precise meaning.

After having introduced some necessary notation, we consider the checks, introduce the "shadow", which is an extraction of the credential mechanism that contains in essence the same information as the simple credential mechanism of §3.4, consider the steps executed by the parti-

cipants of the credential mechanism in more detail, give a proper formulation of the "main condition" by using the shadow, and finally consider the time order in which the steps are executed.

In our model for the credential mechanism we assume that all communication channels are secure against passive and active eavesdropping, and that messages are received at the same moment that they are sent and at the right place. The only essential assumptions in our analysis of the credential mechanism are that the communication channels between individuals and their representatives, and between the organizations and $Z$ are secure. The analysis of the credential mechanism in this paper still holds true if the other assumptions are removed, however this would require uninteresting technical complications in our arguments.

**Notation.** We shall use the same notation as in §2 and §3.1-3.4. In particular, $T=\{0,1,2,\cdots\}$ denotes the time, and $N$ the composite modulus of the underlying RSA-system. We suppose that the set consisting of $N, n, f, m_1, \ldots, m_n, c_1, \ldots, c_K, a, p_1, \ldots, p_L, q_1, \ldots, q_L$ is fixed and known to each participant before the credential mechanism starts. Again we assume that $\phi(N), c_1, \ldots, c_K, a, p_1, \ldots, q_L$ are pairwise coprime and that $p_1, \ldots, q_L$ are primes larger than $\frac{1}{2}n$. Let $\Pi$ be the set of subprotocols introduced in §3.4 and $\mathbb{N}=\{1,2,\cdots\}$. Put (cf. §2.1)

$$Y = F^+(\mathbf{Z}_N^*) \bigcup F(\{1,\ldots,n\}) \bigcup \{\text{true, false}\}.$$

The set of participants $P$ of the credential mechanism consists of the outside world $E$, the signature authority $Z$, the organizations $A_j$ $(j=1,\ldots,L)$, the individuals $i_k$ $(k=1,\ldots,R)$, a set of $LR$ representatives, and the *allocation center* $C$ which is responsible for allocating the representatives to the individuals. We shall discuss later in more detail how this allocation takes place. The message space $M$ is equal to $M' \bigcup M''$ where $M'=\Pi\times\mathbb{N}\times Y$ and $M''$ is an unspecified set, containing the messages which $E$ and $C$ may send or receive.

Let $\mathcal{P}\in\Pi$. For convenience we denote the set $\{\mathcal{P}\}\times\mathbb{N}\times Y\times P_E\times P_E\times T$ by $\mathcal{P}$. Thus $\Pi$ defines a partition of $M'\times P_E\times P_E\times T$ in subprotocols. A step of the form $((\mathcal{P},r,y),\alpha,\beta)$ corresponds to the step of $\mathcal{P}$ in the description of §3.3 which has number $r$ at the left margin, and in which $y$ is sent from $\alpha$ to $\beta$ (or generated by $\alpha$ if $\alpha=\beta$). Messages in which an individual or representative asks for a pseudonym, validator or credential, will be indicated by triples $(\mathcal{P},r,\varnothing)$, for appropriate $\mathcal{P}$ and $r$. Any step of the form $((\mathcal{P},r,y),\alpha,\beta)$ is indicated as "step $r$ of $\mathcal{P}$".

**Checks.** Apart from the security and decision checks described in §3.4, the participants must do some other checks. We assume that at each moment that an individual, $Z$ or an organization receives messages from another participant, it checks if these messages are allowed for the credential mechanism (cf. §2.3, (5)). (For instance, $Z$ checks that the tuple which it receives in step 4 of $(I(i_k,A_j)$ has exactly $2n$ entries, and each individual or organization checks if the time order in which he receives certain messages from some participant is not in conflict with the consistency restriction). These obvious additional checks are also called security checks. Messages which are allowed for the credential mechanism will satisfy all security checks with only the fol-

lowing exception: that two individuals, with representatives $g_{1j}$ and $g_{2j}$ for $A_j$, respectively, do not cheat and by accident generate their numbers such that the checks in step 3 of $II(g_{1j},A_j)$ and $II(g_{2j},A_j)$, result in 'false'.

As before, we assume that a step in $\mathcal{P}$ is executed only if no previously executed step in $\mathcal{P}$ contained a check which resulted in 'false' but allow that the execution of $\mathcal{P}$ is repeated after a security check of the individual involved in $\mathcal{P}$ resulted in 'false'.

**Shadow.** The mapping $\sigma$ on the message space $M$ is defined as follows:

$$\sigma(m)=m \begin{cases} \text{if } m \in M'' \\ \text{or } m=(\mathcal{P},r,y)\in M' \text{ with } y\in\{\text{true, false}\}, \end{cases}$$

$$\sigma(m)=(\mathcal{P},r) \text{ if } m=(\mathcal{P},r,y)\in M' \text{ with } y\notin\{\text{true, false}\}.$$

$\sigma$ is extended to $X=M\times P\times P\times T$ by putting

$$\sigma(m,\alpha,\beta,t) = (\sigma(m),\alpha,\beta,t).$$

For $a\subseteq X$ we put $\sigma(a)=\{\sigma(s):s\in a\}$. We denote $\sigma(X)$ by $\Xi$, and for each subset $\eta$ of $\Xi$ we write $\eta(A,B,U) = \eta\cap(\sigma(M\times A\times B\times U)$ for $A,B\subseteq P$ and $U\subseteq T$. If $S$ is the execution of (an attack on) the credential mechanism then we put $\Sigma=\sigma(S)$ and for any subsets $A$ and $B$ of $P$ and $U$ of $T$ we abbreviate $\sigma(S(A,B,U))$ by $\Sigma(A,B,U)$. $\Sigma$ is called the *shadow* of the (attack on) the credential mechanism.

The shadow is essentially equal to the simple credential mechanism of §3.4, except that it contains values of security checks. But if no participant cheats then these security checks will all result in 'true' with very high probability.

**Individuals, organizations and $Z$.** If a value of $\Sigma$ is given, (i.e. the moments at which the participants execute their messages), then the steps executed by individuals and organizations are completely determined, except for the choices of the pseudonyms $u_k$ in step 1 of $O(i_k)$ (which are not specified), and the uniform choices of the tuples $(r_l,s_l : l=1,\ldots,n)$ in step 3 of $I(i_k,A_j)$ and the sets $\mathbb{S}$ generated in step 6 of $I(i_k,A_j)$ for $1\leqslant k\leqslant R$ and $1\leqslant j\leqslant L$. When saying that at moment $t$, $\alpha$ makes a uniform choice from a finite set $\Gamma$, we implicitly assume that this choice is independent of the other steps executed at or before moment $t$ in which $\alpha$ generated, sent or received a message. We now explain this with the terminology of §2.

Let $\Gamma$ be a finite subset of $Y$. By "$\alpha$ chooses $\gamma$ uniformly from $\Gamma$ at moment $t$ in step $r$ of $\mathcal{P}$ " the following is meant:
let $\Gamma(\mathcal{P},r,\alpha,t)$ denote the choice of $\alpha$ at moment $t$ in step $r$ of $\mathcal{P}$, and let $W(\mathcal{P},r,\alpha,t)$ denote the collection of other steps in which $\alpha$ generated, sent or received a message at or before moment $t$, i.e.

$$W(\mathcal{P},r,\alpha,t) = S(\alpha,P,\leqslant t)\bigcup S(P_\alpha,\alpha,\leqslant t)\setminus(\{(\mathcal{P},r)\}\times\Gamma(\mathcal{P},r,\alpha,t)\times\{(\alpha,\alpha,t)\}).$$

Then for each $\gamma$ in $\Gamma$, and each value $w$ of $W(\mathcal{P},r,\alpha,t)$ we have

$$Pr[\Gamma(\mathcal{P},r,\alpha,t)=\gamma \mid (\mathcal{P},r,\alpha,\alpha,t)\in\Sigma(\alpha,\alpha,t)\ ,\ W(\mathcal{P},r,\alpha,t)=w] = \frac{1}{\#\Gamma}\ . \tag{8}$$

The following two cases are of interest to us:

- $\Gamma=(\mathbf{Z}_N^*)^{2n}$, $\#\Gamma=\phi(N)^{2n}$, $\alpha=i_k$, $\mathcal{P}=I(i_k,A_j)$, $r=3$;
- $\Gamma=\{\mathcal{S}\subset\{1,\ldots,n\}:\#\mathcal{S}=\frac{1}{2}n\}$, $\#\Gamma=\binom{n}{\frac{1}{2}n}$, $\alpha=Z$, $\mathcal{P}=I(i_k,A_j)$, $r=6$.

For the sake of completeness we mention that (8) also holds in case of an attack on the credential mechanism, in which $\alpha$ does not cheat but may receive messages from cheating participants.

**Representatives, allocation center and outside world.** We assume that none of the representatives, the allocation center $C$ or the outside world $E$ will ever cheat. In no execution of (an attack on) the credential mechanism $E$ sends messages to or receives messages from $C$ or the representatives.

During executions of (attacks on) the credential mechanism, $E$ and $C$ send only messages from $M''$ and "neglect" messages outside $M''$, which they may have received during an execution of some attack on the credential mechanism, i.e. the messages they generate or send, are statistically independent of received messages which do not belong to $M''$. Moreover, the allocation center sends messages only to individuals and representatives and neglects messages received from other participants than those.

The representatives belong to a fixed set of cardinality $LR$. We explain how the allocation of representatives to individuals takes place. At moment 1, $C$ allocates a representative to each pair $(i_k,A_j)$, in such a way that different representatives are allocated to different pairs. It is assumed that each allocation has the same probability $(LR)!^{-1}$. At moment 2, $C$ informs each individual, which representatives are allocated to him for communication with the organizations $A_1,\ldots,A_L$, respectively, and informs each representative to which individual it has been allocated and for communication with which organization.

Let $g_j$ be the representative of individual $i_k$ communicating with $A_j$. After $g_j$ has been informed that it has been allocated to $i_k$ for communication with $A_j$, its activities during any execution of (an attack on) the credential mechanism consist only of the following: if $g_j$ receives message $m$ at moment $t$ from a participant $\neq i_k$ then it sends $m$ to $i_k$ at moment $t+1$; whereas if $g_j$ receives $m$ from $i_k$ at moment $t$ then it sends $m$ to $A_j$ at moment $t+1$.

Representatives and allocation center are merely artificial constructions, meant to make the description of the mathematical model somewhat easier, and explain how the credential mechanism looks like from the point of view of the organizations. In general, they will not be used in practical implementations of the credential mechanism.

**Main condition.** $\Sigma(\alpha,P,t)$ describes the communication of participant $\alpha$ with the outside world or the allocation center, at moment $t$, the probability with which $\alpha$ may show pseudonyms, validators or credentials at moment $t$ if $\alpha$ is an individual, or the probability with which $\alpha$ may issue a validator or credential at moment $t$ if $\alpha$ is an organization. The conditional probability of

$\Sigma(\alpha,P,t)$ given $S(\alpha,P,<t)$ and $S(P_\alpha,\alpha,<t)$ describes the freedom of participants to communicate with the outside world, the freedom of individuals to decide whether to ask for or show a pseudonym, validator or credential, and the freedom for $Z$ or the organizations to issue such a pseudonym, validator or credential, at moment $t$. This freedom should be as large as in the simple credential mechanism and hence any restrictions on this freedom should be expressible in the shadow. Thus the main condition can be stated as follows:

for each $\alpha$ in $P$, $t$ in $T$ and execution $s$ of the credential mechanism we have

$$\begin{aligned} &Pr[\sigma(\alpha,P,t)\,|\,s(\alpha,P,<t),s(P_\alpha,\alpha,<t),sec(\alpha,t)] \\ &\quad = Pr[\sigma(\alpha,P,t)\,|\,\sigma(\alpha,P,<t),\sigma(P_\alpha,\alpha,<t),sec(\alpha,t)]\,, \end{aligned} \tag{9}$$

where $\sigma=\sigma(s)$, $sec(E,t)=\varnothing$ and $sec(\alpha,t)$ is the set of values of security checks by $\alpha$ at moment $t$ on messages received from participants other than $E$ if $\alpha\neq E$. If we assume that all security checks result in 'true' (which is extremely likely during executions of the credential mechanism if no participant cheats), then (9) implies that $\Sigma$ is an execution process of a protocol in the sense of §2.

We remark that (9) holds true also for an attack on the credential mechanism in which $\alpha$ does not cheat.

**Time order of steps.** The shadow of the credential mechanism describes the order of the moments at which the steps in the credential mechanism can be executed. We require that steps from the same subprotocol are executed in the same order as described in §3.3, and at consecutive moments. The time order at which steps from different subprotocols $\mathcal{P}$ are executed is subject to the consistency restriction given in §3.4. Other restrictions on the time order of the steps (e.g. those given in §3.4), must imply the main condition (9).

## 4. UNLINKABILITY

An equivalent statement of property 4 mentioned in §1.1 says that the credential mechanism does not reveal any information about which representatives represent which individuals. This property can not be proved in this strict sense. Suppose for instance, that first signature authority $Z$ gives a validator to individual $i_k$ and that later, a representative $g_j$ shows a validator to $A_j$, at a moment at which no other validators have been issued or shown. Then $Z$ and $A_j$ will find out by cooperation, that $g_j$ represents $i_k$. Another situation where information is revealed about the linking between representatives and individuals is the following: suppose that credential $c$ is issued only once, on a pseudonym of representative $g_j$, say, and shown once on a pseudonym of representative $g_h$. Then by cooperation, the issuing and receiving organization will find out that $g_j$ and $g_h$ represent the same individual. We notice that information of the type mentioned above will also be revealed if instead of the credential mechanism of §3.5, the simple credential mechanism described in §3.4 would have been used. Using the model of §3.5 for the credential mechanism, we shall prove that the credential mechanism is optimal in the following sense: all information revealed by the credential mechanism about the relationship between individuals and

representatives is already revealed by the *shadow* of the credential mechanism. As mentioned in §3.5, this shadow is essentially equal to the simple credential mechanism considered in §3.4.

### 4.1. Statement of the result

We shall use the same notation and make the same assumptions as in §§2,3. Thus $P$ is a set consisting of signature authority $Z$, the organizations $A_1, \ldots, A_L$, the individuals $i_1, \ldots, i_R$, $LR$ representatives, the allocation center $C$ and the outside world $E$. Let $J$ be a subset of $P$, consisting of $Z, A_1, \ldots, A_L$ and some of the individuals and let $J_0 = \{i_1, \ldots, i_R\} \setminus J$ be a set of non-cheating individuals. We consider attacks by subsets of $J$ on the credential mechanism. An attack on the credential mechanism is called *safe* for $J$ if it has the following properties (cf. §§2.3,3.5):

- if the messages received by an organization (or $Z$) before moment $t$ from a representative (or individual) were allowed for the credential mechanism, then at moment $t$ that organization ($Z$) sends back messages to that representative (individual) which are also allowed for the credential mechanism;
- no individual sends messages to other individuals or to other individuals' representatives; however, individuals may communicate over the outside world.

Loosely speaking, in safe attacks, cheating individuals, organizations and $Z$ try to hide their cheating from individuals of which they believe that they do not cheat or from representatives of which they believe that they represent a non-cheating individual. An organization can only be sure that some representative represents a cheating individual if he receives messages from that representative which are not allowed for the credential mechanism. The only property of safe attacks which we shall use is, that the messages received by the non-cheating individuals in $J_0$ from participants other than $E$ will satisfy all security checks by these individuals. This is true since in particular the messages received by these individuals' representatives from the organizations are allowed for the credential mechanism. We assume that $Z$ has infinite computational resources, i.e. we make no further restrictions on the choices of $Z$.

Before stating Theorem 1, we recall that the allocation center is denoted by $C$. Thus $\Sigma(C,J_0,2)$ denotes the allocation of representatives at moment 2 to the individuals in $J_0$ (cf. §3.5). We shall abbreviate this by $\Sigma(C,J_0)$. Values $\sigma(C,J_0,2)$ of $\Sigma(C,J_0,2)$ will correspondingly be abbreviated by $\sigma(C,J_0)$. We define $\Theta_t$ as the union of $J_0$ and the set of representatives of $J_0$ which have communicated with some organization, up to moment $t$. Then for each $t>1$ there is a function $\theta_t$ such that $\Theta_t = \theta_t(S(P,J,\leqslant t))$, since $\Theta_t$ contains exactly those representatives not allocated in $\Sigma(C,J,2)$. $S$ will denote the execution process of (an attack on) the credential mechanism, $\Sigma = \sigma(S)$ (cf. shadow in §3.4) and $\Sigma(A,B,U) = \sigma(S(A,B,U))$ for $A,B \subseteq P$ and $U \subseteq T$.

**THEOREM 1.** *Let $J$ be a set consisting of $Z, A_1, \ldots, A_L$ and some of the individuals, and $J_0 = \{i_1, \ldots, i_R\} \setminus J$. Then for each attack on the credential mechanism which is safe for $J$ and in which the individuals in $J_0$ do not cheat, each execution $s$ of this attack, and each $t>1$ we have*

$$Pr[\sigma(C,J_0)\,|\,s(J,P,\leqslant t),s(P,J,,\leqslant t)]$$
$$= Pr[\sigma(C,J_0)\,|\,\Theta_t=\theta_t,\sigma(J,\theta_t \bigcup E,\leqslant t),\sigma(\theta_t \bigcup E,J,\leqslant t)]\,,$$

*where* $\sigma=\sigma(s)$ *and* $\theta_t=\theta_t(s(P,J,\leqslant t))$.

As mentioned above, from $s(J,P,\leqslant t)$ and $s(P,J,\leqslant t)$ it is possible to find out which representatives are allocated to individuals in $J_0$. Theorem 1 tells us that all additional information revealed to $J$ about which representative represents precisely which individual in $J_0$ is already revealed by the shadow of the credential mechanism.

**Remark 1.** In the case, that organizations do not know which individuals are cheating, they can try to find this out by statistically analyzing their set of received messages. We can not state an analogue of Theorem 1 when the set of individuals in $J$ is not fixed, since the collection of attacks on the credential mechanism is not endowed with a probability measure.

**Remark 2.** Suppose that in Theorem 1, $J=\{Z,A_1,\ldots,A_L\}$ and $J_0$ consists of all individuals, and that up to moment $t$ the following happened: for $j=1,\ldots,L$, all individuals got their validators for $A_j$ at the same moment, and all representatives showed their validator to $A_j$ at the same moment; and moreover, no credential was issued or shown. Then $\Theta_t=\theta_t$ where $\theta_t$ consists of all individuals and representatives, and the sets $\Sigma(J,\theta_t,\leqslant t)$ and $\Sigma(\theta_t,J,\leqslant t)$ are independent of $\Sigma(C,J_0)$. Hence in this situation, all information revealed to $J$ about $\Sigma(C,J_0)$ is coming from the sets $\Sigma(J,E,\leqslant t)$ and $\Sigma(E,J,\leqslant t)$, i.e. from $J$'s communication with the outside world.

**Remark 3.** In the statement of Theorem 1, it is essential to assume that the credentials $c_1,\ldots,c_K$ and the exponent $a$ and the primes $p_1,\ldots,q_L$ used in the validating part are all coprime with $\phi(N)$. The individuals have the certainty that this requirement is satisfied if for instance all these numbers are primes larger than $\frac{1}{2}N$. Below we describe a protocol, based on an injective one-way function $h$, in which any individual can convince himself with probability at least $2/3$ that some odd exponent $d$ made public by $Z$ is coprime with $\phi(N)$. That individual can reduce $Z$'s chance of successful cheating by repeating this protocol as many times as he wants. Let $i_k$ be an individual. In step 1, $i_k$ chooses a number $x$ uniformly from $\mathbb{Z}_N^*$, and sends $y:=x^d$ to $Z$. In step 2, $Z$ computes $x'$ with $x'^d=y$ and sends $h_0:=h(x')$ to $i_k$. In step 3, $i_k$ checks if $h(x)=h_0$.

If $d$ is coprime with $\phi(N)$, then the value $x'$ computed by $Z$ is always equal to $x$, and hence $h_0$ is equal to $h(x)$. Suppose that $d$ is not coprime with $\phi(N)$, and let $d_P$, $d_Q$ denote the numbers of solutions of $x^d\equiv 1 \bmod P$, $x^d\equiv 1 \bmod Q$, respectively, where $P$ and $Q$ are the prime factors of $N$. Then there are exactly $d_Pd_Q$ different $x'$ with $x'^d\equiv y \bmod N$. $Z$ knows that $i_k$ must have chosen $x$ in step 1 as one of the $d$-th roots of $y$ but he has no information about which root was precisely chosen by $i_k$. Hence in step 3 $Z$ can do no better than guessing which root was chosen by $i_k$ and the chance that he guesses wrong is $1-(d_Pd_Q)^{-1}$ which is at least $2/3$. By the injectivity of $h$, the chance that $i_k$ will receive a value $h_0$ different from $h(x)$ in step 3 is at least $2/3$. $i_k$ might try to cheat by sending $Z$ a value $y$ in step 1 of which he does not know the $d$-th root. However, if the one-way function $h$ used in step 3 is "good enough", then $i_k$ will not be able to compute the $d$-th root of $y$ from $h_0$.

## 4.2. Lemmas

The idea behind the proof of Theorem 1 is to construct, from the messages sent or received by $Z$ and the organizations and from a given allocation of representatives to individuals, a set of tuples $(r_l,s_l\colon l=1,\ldots,n)$, chosen by the individuals in step 3 of each subprotocol $I(i_k,A_j)$, and argue that the sets of tuples constructed from different allocations have equal probability. In this section we state and prove two lemmas needed in the proof.

Let $J$ consist of $Z,A_1,\ldots,A_L$ and some of the individuals, and let $J_0=\{i_1,\ldots,i_R\}\setminus J$. We fix an attack on the credential mechanism which is safe for $J$ and in which the individuals in $J_0$ do not cheat, and denote the execution process of this attack by $S$. Further we put $\Sigma=\sigma(S)$. Up to now, by an execution we just meant an arbitrary value of the execution process of the attack we are considering, which can be any subset of $X=M\times P\times P\times T$. In the sequel we shall restrict ourselves to *proper* executions, i.e. executions $s$ with the following properties:

- the set of elements of $s$ belonging to the subprotocols in which individuals in $J_0$ or their representatives are involved satisfies the description of these subprotocols in §3.3; in particular, in each step 3 of $I(i_k,A_j)$ with $i_k\in J_0$ a tuple $(r_l,s_l\colon l=1,\ldots,n)\in(\mathbf{Z}_N^*)^{2n}$ is chosen, and in each of the subprotocols involving individuals in $J_0$ or their representatives, the messages are computed as prescribed in §3.3, and the steps are executed in the order corresponding with the description of §3.3 and at consecutive moments;
- $s$ satisfies the conditions imposed on the messages generated, sent or received by $E$, $C$ and the representatives as described in §3.5;
- the time order in which the steps in $s$, involving an individual in $J_0$ or one of his representatives, are executed, is subject to the consistency restriction of §3.4.

In attacks which are safe for $J$ and in which the individuals in $J_0$ do not cheat, executions which are not proper have always probability 0. There may be proper executions with probability 0.

In Lemma 1 below we compute the probability of a proper execution. We put $J'_0=J_0\cup\{E\}$. For any proper execution $s$ with $\sigma=\sigma(s)$, and any subset $U$ of $T$, we denote by $\kappa(\sigma(J_0,P,U))$ the number of all steps 3 of $I(i_k,A_j)$ executed during $U$, for $i_k\in J_0$ and $1\leqslant j\leqslant L$. It is clear that this number depends indeed only on $\sigma(J_0,P,U)$.

**Lemma 1.** *For every $t>0$ there are functions $A_t$ and $B_t$ such that for each proper execution $s$,*

$$\begin{aligned}Pr[S(P,P,\leqslant t)] = \phi(N)^{-2n\kappa(\sigma(J_0,P,\leqslant t))}\\ \times A_t(\sigma(J'_0,P,\leqslant t),\sigma(C,J_0),\theta_t,\sigma(J,\theta_t\cup E,\leqslant t))\\ \times B_t(s(J,P,\leqslant t),s(P,J,\leqslant t))\,,\end{aligned}$$

*where $\sigma=\sigma(s)$ and $\theta_t=\theta_t(s(P,J,\leqslant t))$ is the value for $\Theta_t$.*

**Proof.** In the proof of this lemma only, undefined conditional probabilities will be given the value 1. We shall prove Lemma 1 by induction on $t$. We start with $t=0$.

The statement of Lemma 1 trivially holds true for $t=0$ by taking $\kappa(\sigma(J_0,P,0)=0$ and letting $A_0$ and $B_0$ be functions identically equal to 1. Suppose that Lemma 1 has been proved for moment $t-1$ where $t>0$. (induction hypothesis). We proceed to prove Lemma 1 for moment $t$.

Let $s$ be a proper execution. For convenience we put

$$Pr(t)=Pr[s(P,P,\leqslant t)]\ ,\ Pr(t-1)=Pr[s(P,P,<t)]\ .$$

Let $R$ denote the set of representatives. By (1) (cf. §2.1) we have

$$\begin{aligned} Pr(t) &= Pr[s(J'_0,P,\leqslant t),s(C,P,\leqslant t),s(R,P,\leqslant t),s(J,P,\leqslant t)] \\ &= P_1P_2P_3P_4Pr(t-1)\ , \end{aligned} \tag{10}$$

where

$$\begin{aligned} P_1 &= Pr[s(J'_0,P,t)\,|\,s(J'_0,P,<t],s(C,P,\leqslant t),s(R,P,\leqslant t),s(J,P,\leqslant t))\ , \\ P_2 &= Pr[s(C,P,t)\,|\,s(J'_0,P,<t),s(C,P,<t),s(R,P,\leqslant t),s(J,P,\leqslant t)]\ , \\ P_3 &= Pr[s(R,P,t)\,|\,s(J'_0,P,<t),s(C,P,<t),s(R,P,<t),s(J,P,\leqslant t)]\ , \\ P_4 &= Pr[s(J,P,t)\,|\,s(J'_0,P,<t),s(C,P,<t),s(R,P,<t),s(J,P,<t)]\ . \end{aligned}$$

Note that (10) also holds true if some of the conditional probabilities on the right-hand side are not defined. Moreover, if one of the conditional probabilities in (10) is not defined then one of the other factors in the right-hand side of (10) is 0. Therefore, (10) remains true if we replace an undefined conditional probability by any value we like. To the defined conditional probabilities we may apply (4) (cf. §2.2) with the partition $J'_0,\{C\},R$ and $J$ of $P$. Thus we obtain

$$Pr(t)=P'_1P'_2P'_3P'_4Pr(t-1)\ , \tag{11}$$

where

$$\begin{aligned} P'_1 &= Pr[s(J'_0,P,t)\,|\,s(J'_0,P,<t),s(P,J'_0,<t)]\ , \\ P'_2 &= Pr[s(C,P,t)\,|\,s(C,P,<t),s(P,C,<t)]\ , \\ P'_3 &= Pr[s(R,P,t)\,|\,s(R,P,<t),s(P,R,<t)]\ , \\ P'_4 &= Pr[s(J,P,t)\,|\,s(J,P,<t),s(P,J,<t)]\ . \end{aligned}$$

Because of the relationships between messages sent and received by the representatives, which hold also for our proper execution $s$, we have $P'_3=1$. Moreover, if $t=1$ we have $P'_2=(LR)!^{-1}$, since each allocation is equally likely, if $t=2$ then $P'_2=1$ since the messages sent by the allocation center to the individuals and representatives are determined by the allocation at moment 1, and if $t>2$ then also $P'_2=1$, since $s(C,P,t)=\varnothing$ with probability 1. (Here we used that $s$ satisfies the conditions on the messages generated and sent by $C$). By combining these facts we obtain that there is a function $C_t$ such that

$$P'_2P'_3P'_4=C_t(s(J,P,\leqslant t),s(P,J,\leqslant t))\ . \tag{12}$$

We now consider $P'_1$. Suppose for the moment that $P'_1$ is defined. All elements of

$s(J'_0,P,t)$ are of the form $x=(m,\alpha,\beta,t)$, where $\alpha \in J'_0$. If $m \in M''$ or $m=(\mathcal{P},r,y) \in M'$ and $y \in \{\text{true, false}\}$, then $x \in \sigma(J'_0,P,t)$. Otherwise we have $m=(\mathcal{P},r,y)$, where $(\mathcal{P},r,\alpha,\beta)$ belongs to $\sigma(J'_0,P,t)$ and $y$ is either a tuple $(r_l,s_l\colon l=1,\ldots,n)$ chosen in step 3 of some $I(i_k,A_j)$, or can be derived from a tuple previously chosen in step 3 of some $I(i_k,A_j)$ and from messages previously received by $J_0$, by using the computations in the credential mechanism described in §3.3. It follows that $S(J'_0,P,t)$ is completely determined by $\Sigma(J'_0,P,t)$, $B(t)$, $S(J'_0,P,<t)$ and $S(P,J'_0,<t)$, where $B(t)$ is the set of tuples chosen at moment $t$ in step 3 of some $I(i_k,A_j)$, for $i_k \in J_0$. Let $b(t)$ be the value of $B(t)$ corresponding to $s(J'_0,P,t)$. Then

$$P'_1=Pr[b(t),\sigma(J'_0,P,t)\,|\,s(J'_0,P,<t),s(P,J'_0,<t)]=P'_{11}P'_{12}\,, \tag{13}$$

where

$$P'_{11}=Pr[b(t)\,|\,\sigma(J'_0,P,t),s(J'_0,P,<t),s(P,J'_0,<t)]\,,$$
$$P'_{12}=Pr[\sigma(J'_0,P,t)\,|\,s(J'_0,P,<t),s(P,J'_0,<t)]\,.$$

$b(t)$ contains exactly $\kappa(\sigma(J'_0,P,t))$ tuples $(r_l,s_l\colon l=1,\ldots,n)$. By assumption, the distributions of these tuples are uniform on $(\mathbf{Z}_N^*)^{2n}$ and independent of each other and of $S(J'_0,P,<t)$ and $S(P,J'_0,<t)$. By repeatedly applying (8) to these tuples and using (1) we obtain

$$P'_{11}=\phi(N)^{-2n\kappa(\sigma(J_0,P,t))}\,, \tag{14}$$

provided that $P'_{11}$ is defined. If $P'_{11}$ is not defined then $P'_{12}=0$, hence (13) still holds true if we replace $P'_{11}$ by the right-hand side of (14). Since we are considering a safe attack, the security checks by individuals in $J_0$ on messages received from participants other than $E$ will be satisfied with probability 1. Therefore we may apply main condition (9) for each $\alpha$ in $J'_0$ without the stochastic variable $sec(\alpha,t)$ on both sides of the equality. By combining (9) for each $\alpha$ in $J'_0$ with (1), we obtain

$$P'_{12}=Pr[\sigma(J'_0,P,t)\,|\,\sigma(J'_0,P,<t),\sigma(P,J'_{0,}<t)]\,. \tag{15}$$

We note that $E$ receives messages only from $J_0,E$ and $J$, that $\sigma(J'_0,E,<t)$ is contained in $\sigma(J'_0,P,<t)$, while $\sigma(J,E,<t)$ is contained in $\sigma(J,\theta_t \bigcup E,<t)$. $J_0$ receives messages only from $C,E,Z$ and the representatives in $\theta_t$. $\sigma(E,J_0,<t)$ and $\sigma(Z,J_0,<t)$ are contained in $\sigma(J'_0,P,<t)$ and $\sigma(J,\theta_t \bigcup E,<t)$, respectively. Moreover, from the relationship between messages sent, and messages received by representatives (which holds for proper executions), it follows that $\sigma(\theta_t,J_0<t)$ is uniquely determined by $\sigma(C,J_0)$ and $\sigma(J,\theta_t,<t)$. By combining these facts we conclude that $\sigma(P,J'_0,<t)$ can be expressed as a function in $\sigma(J'_0,P,<t)$, $\sigma(C,J_0)$, $\theta_t$ and $\sigma(J,\theta_t \bigcup E,<t)$. A combination of this with (13), (14) and (15) yields that there is a function $D_t$ with

$$P'_1=\phi(N)^{-2n\kappa(\sigma(J_0,P,t))}D_t(\sigma(J'_0,P,\leqslant t),\sigma(C,J_0),\theta_t,\sigma(J,\theta_t \bigcup E,<t))\,.$$

By combining this with (12) and (11) we obtain

$$Pr(t) = \phi(N)^{-2n\kappa(\sigma(J_0,P,t))} D_t(\sigma(J'_0,P,\leqslant t)C_t(s(J,P,\leqslant t),s(P,J,\leqslant t))\times Pr(t-1)\,.$$

Note that this is true also if $P'_1$ is not defined. Together with the induction hypothesis this proves Lemma 1 for moment $t$. □

Let $F_1,\ldots,F_r$ be functions of $S$. We say that the values $s_1,\ldots,s_r$ of $F_1(S),\ldots,F_r(S)$, respectively, *come from the same proper execution* if there is a proper execution $s$ such that $s_1=F_1(s),\ldots,s_r=F_r(s)$. Lemma 2 gives the number of possibilities for the messages generated, sent or received by the individuals in $J_0$, given only the moments at which the individuals in $J_0$ generate, send or receive their messages, the allocation of representatives to $J_0$, and the steps involving the members of $J$.

**Lemma 2.** *There is a function* $\lambda$ *with the following property: for each* $t>0$, *and all values* $\sigma_0$ *of* $\Sigma(J'_0,P,\leqslant t)$, $\sigma_{C0}$ *of* $\Sigma(C,J_0)$, $\theta_t$ *of* $\Theta_t$, $s_{JP}$ *of* $S(J,P,\leqslant t)$, $s_{PJ}$ *of* $S(P,J,\leqslant t)$, $\sigma_{J*}$ *of* $\Sigma(J,\Theta_t\cup E,\leqslant t)$, *and* $\sigma_{*J}$ *of* $\Sigma(\Theta_t\cup E,J,\leqslant t)$, *such that*

$$Pr[s_{JP},s_{PJ},\theta_t,\sigma_{J*},\sigma_{*J}]>0$$

*and*

$\sigma_0,\sigma_{C0},\theta_t,\sigma_{J*},\sigma_{*J}$ *come from the same proper execution,*

*there are exactly* $\phi(N)^{2n\kappa(\sigma_0)-\lambda(n,\theta_t,\sigma_{*J})}$ *values* $s_0$ *of* $S(J'_0,P,\leqslant t)$ *such that* $s_0,\sigma_0$, $\sigma_{C0}$, $\theta_t$, $s_{JP}$, $s_{PJ}$, $\sigma_{J*}$ *and* $\sigma_{*J}$ *come from the same proper execution.*

**Proof.** In this proof we shall often refer to the description of the credential mechanism in §3.3, and the reader is advised to consult this. The values $s_0$ of $S(J'_0,P,\leqslant t)$ we are looking for, consist of tuples $x=(m,\alpha,\beta,u)$ with $m\in M$, $\alpha\in J'_0$, $\beta\in P$ and $u\leqslant t$. If $m\in M''$ or $m=(\mathcal{P},r,y)\in M'$ with $y\in\{\text{true, false}\}$ then $x\in\sigma_0$. For the remaining tuples $x$ we have $\alpha\neq E$, $\beta\neq E$ and $m=(\mathcal{P},r,y)\in M'$, where each $\sigma(x)=(\mathcal{P},r,\alpha,\beta,u)$ is contained in $\sigma_0$ and the set of values $y$ must satisfy the constraints imposed by the description of the credential mechanism in §3.3, the given allocation of representatives, and the steps in which the members in $J$ generated, sent or received their messages before moment $t$. Our purpose is to count the number of possibilities for the set of values $y$. From the description of the credential mechanism in §3.3 it follows that each $y$ is either equal to one of the tuples $(r_l,s_l\colon l=1,\ldots,n)$ generated in the steps 3 of $I(i_k,A_j)$ up to moment $t$, or can be derived from these tuples and the messages received by $J_0$ up to moment $t$ from $Z$ or their representatives, by using the computations prescribed in the credential mechanism. But in proper executions, the messages received by $J_0$ can be determined from the allocation $\sigma_{C0}$ and $s_{JP}$. Hence the number of possibilities for the set of values $y$ is equal to the number of possibilities for the set of tuples $(r_l,s_l\colon l=1,\ldots,n)$ generated in each step 3 of $I(i_k,A_j)$ up to moment $t$ for $i_k\in J_0$ and $1\leqslant j\leqslant L$. We shall prove that this number is equal to $\phi(N)^{2n\kappa(\sigma_0)-\lambda(n,\theta_t,\sigma_{*J})}$ where

$$\lambda(n,\theta_t,\sigma_{*J})=n\lambda_1+\tfrac{1}{2}n\lambda_2+2\lambda_3+(\tfrac{1}{2}n-2)\lambda_4\,, \tag{16}$$

and $\lambda_1$ is the number of steps 5 of $I(i_k,A_j)$, $\lambda_2$ the number of steps 8 of $I(i_k,A_j)$, $\lambda_3$ the number

of steps 2 of $II(g_j,A_j)$, and $\lambda_4$ the number of steps 5 of $II(g_j,A_j)$, for all $i_k \in J_0$, $g_j \in \theta_t$, and organizations $A_j$, which are executed up to moment $t$. Note that all these steps are contained in $\sigma_{*J}$.

Let $i_k \in J_0$ and $g_j$ the representative allocated to $i_k$ for communication with $A_j$. Suppose that at or before moment $t$, step 3 of $I(i_k,A_j)$ has been executed (this can be derived from $\sigma_0$), and let $\rho=(r_l,s_l: l=1,\ldots,n)$ be the tuple chosen in this step. We have to count the number of values for $\rho$ such that $\rho$, $\sigma_0$, $\sigma_{C0}$, $\ldots$, $\sigma_{*J}$ come from the same proper execution. Each step $v$, executed up to moment $t$, in which $Z$ or an organization receives a message from an individual in $J_0$ or one of its representatives (which is contained in $s_{PJ}$), imposes certain constraints on $\rho$, which will reduce the number of possibilities for $\rho$. Without any of these constraints, there are $\phi(N)^{2n}$ possibilities for $\rho$.

If $v$ is step 5 of $I(i_k,A_j)$, then $Z$ receives numbers $a_{kl}$ $(l=1,\ldots,n)$, which should be equal to $f(u_k(m_l r_l^a)^{b_j})s_l^{p_j q_j}$. These relationships reduce the number of possibilities for $\rho$ by a factor $\phi(N)$. Indeed, since $p_j q_j$ is coprime with $\phi(N)$, each $r_l$ determines a unique $s_l$ such that these relationships are satisfied.

If $v$ is step 8 of $I(i_k,A_j)$ then $Z$ receives $r_l,s_l$ with $l \in \mathbb{S}$, where $\mathbb{S}$ is the set sent by $Z$ to $i_k$ in step 7. As remarked before, each $s_l$ is determined by $a_{kl}$ and $r_l$ and does not impose additional conditions on $\rho$. Obviously, the released values $r_l$ reduce the number of possibilities for $\rho$ by a factor $\phi(N)^{\frac{1}{2}n}$.

If $v$ is step 2 of $II(g_j,A_j)$, then $A_j$ receives $u_{g_j}$, which must be equal to $u_k^{-b_j}\tilde{r}_{\sigma(1)}^{-b_j}$, and $w_{g_j}$, which must be equal to $u_k^{\frac{1}{2}n-1}(\prod_{l=2}^{\frac{1}{2}n}\tilde{r}_{\sigma(l)})^{b_j}$, where $\sigma$ is the permutation defined in the description of $II(g_j,A_j)$ in §3.3, and $\tilde{r}_{\sigma(l)}=m_{\sigma(l)}r_{\sigma(l)}^a$ for $l=1,\ldots,\frac{1}{2}n$. Since both $a$ and $b_j$ are coprime with $\phi(N)$, $r_{\sigma(1)}$ and $\prod_{l=2}^{\frac{1}{2}n} r_{\sigma(l)}$ are uniquely determined by $u_{g_j}$ and $w_{g_j}$. This reduces the number of possibilities for $\rho$ by a factor $\phi(N)^2$.

Finally, if $v$ is step 5 of $II(g_j,A_j)$ then $A_j$ receives $\tilde{v}_{g_j}$, and also numbers $t_l$ $(l=2,\ldots,\frac{1}{2}n)$ which should be equal to $\tilde{r}_{\sigma(l)} / \tilde{r}_{\sigma(1)}$, respectively. From this it is possible to determine uniquely a set of values $r_l$, not shown to $Z$ in step 8 of $I(i_k,A_j)$. Since each of these $r_l$ determines a unique $s_l$, this leaves us only one possibility for $\rho$. In other words, the number of possibilities for $\rho$ is reduced by a factor $\phi(N)^{\frac{1}{2}n-2}$.

Other steps $v$ in which $Z$ or the organizations received messages up to moment $t$ do not further reduce the number of possibilities for $\rho$. Thus for each tuple $\rho$ generated in step 3 of some $I(i_k,A_j)$ we have a specified number of possibilities. We obtain (16) by taking the product of all these numbers of possibilities, over all steps 3 of all $I(i_k,A_j)$ with $i_k \in J_0$, executed at or before moment $t$. □

### 4.3. Proof of Theorem 1

We shall prove that for each $t>1$ there are functions $E_t$ and $F_t$ such that for all values $\sigma_{C0}$ of $\Sigma(C,J_0)$, $\theta_t$ of $\Theta_t$, $s_{JP}$ of $S(J,P,\leqslant t)$, $s_{PJ}$ of $S(P,J,\leqslant t)$, $\sigma_{J^*}$ of $\Sigma(J,\Theta_t \cup E,\leqslant t)$ and $\sigma_{*J}$ of $\Sigma(\Theta_t \cup E,J,\leqslant t)$, with $Pr[\theta_t,s_{JP},s_{PJ},\sigma_{J^*},\sigma_{*J}] = Pr[s_{JP},s_{PJ}]>0$ we have

$$Pr[\sigma_{C0},s_{JP},s_{PJ}] = \phi(N)^{-\lambda(n,\theta_t,\sigma_{*J})} E_t(\sigma_{C0},\theta_t,\sigma_{J^*},\sigma_{*J}) F_t(s_{JP},s_{PJ}) . \tag{17}$$

This suffices. For by summing over all $\sigma_{C0}$ we obtain that there is a function $G_t$ with

$$Pr[s_{JP},s_{PJ}] = \phi(N)^{-\lambda(n,\theta_t,\sigma_{*J})} G_t(\theta_t,\sigma_{J^*},\sigma_{*J}) F_t(s_{JP},s_{PJ}) .$$

Hence $G_t(\theta_t,\sigma_{J^*},\sigma_{*J})\neq 0$ and $F_t(s_{JP},s_{PJ})\neq 0$. This gives

$$Pr[\sigma_{C0} \mid s_{JP},s_{PJ}] = \frac{E_t(\sigma_{C0},\theta_t,\sigma_{J^*},\sigma_{*J})}{G_t(\theta_t,\sigma_{J^*},\sigma_{*J})} = Pr[\sigma_{C0} \mid \sigma_{J^*},\sigma_{*J},\theta_t] .$$

We now prove (17). We fix $\sigma_{C0},\theta_t,s_{JP}$, $s_{PJ},\sigma_{J^*},\sigma_{*J}$ with $Pr[\theta_t,s_{JP},s_{PJ},\sigma_{J^*},\sigma_{*J}]\neq 0$ and assume that $Pr[\sigma_{C0},\theta_t,\sigma_{J^*},\sigma_{*J}]\neq 0$ which is no restriction. We have

$$Pr[\sigma_{C0},s_{JP},s_{PJ}] = \sum_{\sigma_0} \left[ \sum_{s_0} Pr[s_0,s_{JP},s_{PJ},\sigma_{C0}] \right] , \tag{18}$$

where the outer sum is taken over all values $\sigma_0$ of $\Sigma(J'_0,P,\leqslant t)$ such that $\sigma_0,\sigma_{C0},\theta_t,\sigma_{J^*},\sigma_{*J}$ come from the same proper execution, and the inner sum over all values $s_0$ of $S(J'_0,P,\leqslant t)$ such that $s_0$, $\sigma_0$, $\sigma_{C0}$, $\theta_t$, $s_{JP}$, $s_{PJ}$, $\sigma_{J^*}$, and $\sigma_{*J}$ come from the same proper execution. From $s_0$, $\sigma_{C0}$, $s_{JP}$, and $s_{PJ}$ it is possible to derive all steps executed by $C$ and the representatives. Hence if $s_0$, $\sigma_0$, $\sigma_{C0}$, $\theta_t$, $s_{JP}$, $s_{PJ}$, $\sigma_{J^*}$, and $\sigma_{*J}$ come from the same proper execution $s$, then $Pr[s_0,s_{JP},s_{PJ},\sigma_{C0}]= Pr[s(P,P,\leqslant t)]$. By Lemma 1 this implies that

$$Pr[s_0,s_{JP},s_{PJ},\sigma_{C0}] = \phi(N)^{-2n\kappa(\sigma_0)} A_t(\sigma_0,\sigma_{C0},\theta_t,\sigma_{J^*}) B_t(s_{JP},s_{PJ}) .$$

Together with Lemma 2 this gives

$$\sum_{s_0} Pr[s_0,s_{JP},s_{PJ},\sigma_{C0}] = \phi(N)^{-\lambda(n,\theta_t,\sigma_{*J})} A_t(\sigma_0,\sigma_{C0},\theta_t,\sigma_{J^*}) B_t(s_{JP},s_{PJ}) .$$

By taking the sum over all $\sigma_0$ for which $\sigma_0$, $\sigma_{C0}$, $\theta_t$, $\sigma_{J^*}$ and $\sigma_{*J}$ come from the same proper execution, we obtain

$$\sum_{\sigma_0} \left[ \sum_{s_0} Pr[s_0,s_{JP},s_{PJ},\sigma_{C0}] \right] = \phi(N)^{-\lambda(n,\theta_t,\sigma_{*J})} E_t(\sigma_{C0},\theta_t,\sigma_{J^*},\sigma_{*J}) B_t(s_{JP},s_{PJ})$$

for some function $E_t$. Together with (18) this implies (17). This completes the proof of Theorem 1. □

## 5. THE FORMAL CREDENTIAL MECHANISM

In this section we describe a formal credential mechanism which is similar to that of §3 except that it is based on an "ideal RSA-cryptosystem" of which the underlying message space is a free multiplicative module over the rational numbers with denominator coprime with $\phi(N)$ and the one-way function maps this message space on multiplicatively independent elements of this space. In §§6,7 we shall analyze this formal credential mechanism for protection of organizations against cheating individuals.

### 5.1. Description of the underlying message space

In order to give a proper description of the underlying cryptosystem we have to introduce some notions about free multiplicative modules.

For any set $\mathcal{V}$ and any integral domain $\mathcal{R}$, we denote by $\mathcal{R}[\mathcal{V}]$ the set of all finite formal products

$$V_1^{\xi_1} \cdots V_t^{\xi_t}$$

where $\xi_1, \ldots, \xi_t \in \mathcal{R}$ and $V_1, \ldots, V_t$ are different elements of $\mathcal{V}$. The empty product is denoted by 1. We shall identify two formal products $V_1^{\xi_1} \cdots V_t^{\xi_t}$ and $W_1^{\eta_1} \cdots W_s^{\eta_s}$ if and only if there is an $r$ with $r \leqslant t$ and $r \leqslant s$ such that after reordering the terms of both products, $\xi_i = 0$ for $r < i \leqslant t, \eta_i = 0$ for $r < i \leqslant s$ and $V_i = W_i$ and $\xi_i = \eta_i$ for $1 \leqslant i \leqslant r$. $\mathcal{R}[\mathcal{V}]$ is a free $\mathcal{R}$-module, of which the addition is the multiplication of formal products, defined by adding the exponents, and scalar multiplication is raising a formal product to a power in $\mathcal{R}$, i.e. multiplying the exponents of that product with that power.

We shall recursively construct a free $\mathcal{R}$-module which contains $\mathcal{V}$ and which is closed under application of some "formal one-way function". Put

$$\mathcal{R}_1 = \mathcal{R}[\mathcal{V}], \quad \mathcal{F}_1 = \{F_X : X \in \mathcal{R}_1\};$$
$$\mathcal{R}_i = \mathcal{R}[\mathcal{R}_{i-1} \cup \mathcal{F}_{i-1}], \quad \mathcal{F}_i = \{F_X : X \in \mathcal{R}_i \setminus \mathcal{R}_{i-1}\} \text{ for } i = 2,3,4, \cdots$$

and assume that $F_X \neq F_Y$ if $X \neq Y$ and $\mathcal{F}_i \cap \mathcal{R}_i = \varnothing$ for $i = 1,2,3, \cdots$. Put $\mathcal{F} = \bigcup_{i=1}^{\infty} \mathcal{F}_i$ and define the function $F : \mathcal{R}[\mathcal{V} \cup \mathcal{F}] \to \mathcal{F}$ by $F(X) = F_X$. The module $\mathcal{R}[\mathcal{V} \cup \mathcal{F}]$ endowed with the function $F$ just defined is denoted by $\mathcal{R}[F, \mathcal{V}]$. Different choices of the sets $\mathcal{F}_i$ will lead to isomorphic modules $\mathcal{R}[F, \mathcal{V}]$. If $\mathcal{C}$ is a subset of $\mathcal{R}[F, \mathcal{V}]$ we denote by $\mathcal{R}[F, \mathcal{C}]$ the smallest $\mathcal{R}$-submodule of $\mathcal{R}[F, \mathcal{V}]$ which contains $\mathcal{C}$ and is closed under application of $F$.

To the numbers $u_1, \ldots, u_R$ issued to the individuals $i_1, \ldots, i_R$ by $Z$ in the credential mechanism, and to $m_1, \ldots, m_n$, chosen by $Z$ during the initialization of the credential mechanism, we associate formal variables $U_1, \ldots, U_R, M_1, \ldots, M_n$, respectively. Also other formal variables $H_1, \ldots, H_T$ are introduced, which correspond to numbers chosen by the individuals

themselves. Let

$$\tilde{\mathbf{Q}} = \{\frac{a}{b} : a,b \in \mathbf{Z},\ \gcd(b,\phi(N))=1\},$$
$$\mathcal{K}=\{U_1,\ldots,U_R,M_1,\ldots,M_n,H_1,\ldots,H_T\},$$
$$\mathfrak{L}=\tilde{\mathbf{Q}}[F,\mathcal{K}],$$

where $F$ is a formal one-way function as described above.

*F-homomorphisms* as defined below establish a relationship between the pairs $(\mathfrak{L},F)$ and $(\mathbf{Z}_N^*,f)$. A mapping $\psi : \mathfrak{L}\rightarrow\mathbf{Z}_N^*$ is called an F-homomorphism if it has the following properties:

$\psi$ is a homomorphism with respect to multiplication;
$\psi(F(W))=f(\psi(W))$ for $W\in\mathfrak{L}$;
$\psi(W^{\frac{a}{b}})=\left[\psi(W)^a\right]^{\bar{b}}$ for $W\in\mathfrak{L},a\in\mathbf{Z},b\in\mathbf{Z}$ with $\gcd(b,\phi(N))=1$;
$\psi(U_k)=u_k$ for $k=1,\ldots,R$, $\psi(M_l)=m_l$ for $l=1,\ldots,n$.

Thus an F-homomorphism is uniquely determined by its images in $H_1,\ldots,H_T$.

## 5.2. Computations on $\mathfrak{L}$

$\mathfrak{L}$ is closed under the following operations: multiplications, multiplicative inversions, applications of $F$, and *taking roots*, i.e. raising to powers $a^{-1}$ where $a$ is an integer, coprime with $\phi(N)$. We shall endow $\mathfrak{L}$ with a computational model, based on these operations, which is used in protocols of which (part of) the transmitted messages are elements of $\mathfrak{L}$.

A *computation* on $\mathfrak{L}$ is a repeated application of multiplications, multiplicative inversions, and $F$, (but *not* taking roots), to elements of $\mathfrak{L}$. If $\mathfrak{D}\subseteq\mathfrak{L}$ then $X\in\mathfrak{L}$ is said to be *computable* from $\mathfrak{D}$ if it can be obtained by applying a computation to elements of $\mathfrak{D}$. Thus the set of elements of $\mathfrak{L}$ computable from $\mathfrak{D}$ is equal to $\mathbf{Z}[F,\mathfrak{D}]$. By assumption, the elements of $\mathcal{K}=\{U_1,\ldots,U_R,M_1,\ldots,M_n,H_1,\ldots,H_T\}$ are computable.

An *extended computation* on $\mathfrak{L}$ is a repeated application of multiplications, multiplicative inversions, $F$, and also taking roots, to elements of $\mathfrak{L}$. Each element of $\mathfrak{L}$ can be obtained from $\mathcal{K}$ by extended computations.

Consider a protocol in which $\mathfrak{L}$ is (part of) the message space, such that each participant may apply all possible computations to the elements of $\mathcal{K}$ and to the messages which it received during an execution of the protocol, but only a few distinguished participants are allowed to do extended computations. Let $\alpha$ be a participant which is not allowed to do extended computations, and let $\mathfrak{D}(<t)$ be the stochastic set of elements of $\mathfrak{L}$, received by $\alpha$ before $t$, which are not computable from $\mathcal{K}$. Then at moment $t$, $\alpha$ can compute each element of the set $\mathbf{Z}[F,\mathcal{K}\bigcup\mathfrak{D}(<t)]$. Obviously, this set is stochastic and might grow for increasing $t$.

Let $\psi$ be an $F$-homomorphism, and let $\mathfrak{D}$ be a subset of $\mathfrak{L}$, containing $\mathcal{K}$. Suppose that

some participant $\alpha$ has received all messages in $\psi(\mathfrak{N})$ during the execution of some protocol, of which $\mathbf{Z}_N^*$ is (part of) the message space. If $\alpha$ can break RSA for the modulus $N$, then he can, in principle, compute $\psi(A)$ for each $A$ in $\mathfrak{T}$. If $\alpha$ can not break this RSA-system then in principle he can still compute $\psi(A)$ for each $A$ in $\mathfrak{T}$ which is computable from $\mathfrak{N}$, by applying multiplications and multiplicative inversions in $\mathbf{Z}_N^*$ and $f$ to the elements of $\psi(\mathfrak{N})$. As far as we know, the following question —which should be stated more precisely— is still open:

> are there a modulus $N$ and a one-way function $f:\mathbf{Z}_N^*\to\mathbf{Z}_N^*$, such that for each F-homomorphism $\psi$ and each $A$ in $\mathfrak{T}$ not computable from $\mathfrak{N}$, computing $\psi(A)$ from $\psi(\mathfrak{N})$ is as difficult as breaking RSA for the modulus $N$?

## 5.3. The formal credential mechanism

We shall use the same notation as in the previous sections. The formal credential mechanism will have the same set of participants $P$ as the credential mechanism of §3, consisting of the signature authority $Z$, the organizations $A_1,\ldots,A_L$, the individuals $i_1,\ldots,i_R$, the individuals' representatives, the outside world $E$ and the allocation center $C$. As before, the numbers $\phi(N)$, $c_1,\ldots,c_K, a, p_1,\ldots,p_L, q_1,\ldots,q_L$ are pairwise coprime and $p_1,\ldots,q_L$ are primes larger than $\frac{1}{2}n$, where $n>4$ is the security parameter in the validating part. The message space $M$ will be defined in the same way as in §3.5, except that $\mathbf{Z}_N^*$ is replaced by $\mathfrak{T}$. Thus $M=M'\cup M''$, where $M'=\Pi\times\mathbb{N}\times Y$ with $Y=F^+(\mathfrak{T})\cup F(\{1,\ldots,n\})\cup\{\text{true, false}\}$ and $M''$ is the set of messages sent or received by $E$ and $C$. $A\in\mathfrak{T}$ is said to be contained in a message $m$ if $m=(\mathfrak{P},r,y)\in M'$ and $A$ is an entry of $y$. We say that participant $\alpha$ generates (sends, receives) $A\in\mathfrak{T}$ in step $r$ of $\mathfrak{P}$ if $\alpha$ generates (sends, receives) the message $(\mathfrak{P},r,A)$. Again the time will be modelled as a set of discrete moments, $T=\{0,1,2,\cdots\}$. Again we put $X=M\times P\times P\times T$. For each $x=(m,\alpha,\beta,t)\in X$, we let $C(x)$ denote the set of elements of $\mathfrak{T}$ contained in $m$. In particular, $C(x)=\varnothing$ if $m\in M''$. If $a$ is a subset of $X$ then we put $C(a)=\bigcup_{x\in a}C(x)$.

We postulate that all participants of the credential mechanism can apply all computations to elements of $\mathfrak{K}$ and elements of $\mathfrak{T}$ which they received during an execution of (an attack on) the formal credential mechanism; only signature authority $Z$ can do extended computations. The computational abilities of each participant $\alpha$ of the formal credential mechanism can be expressed in terms of its collection of choices $\mathcal{C}_\alpha$ (cf. §2.3). The outside world, allocation center and representatives will all have a unique choice, satisfying the same conditions as in §3.5. $\mathcal{C}_Z$ contains all choices satisfying property (2) with $\alpha=Z$. (cf. §2.2). If $\alpha\in\{i_1,\ldots,i_R,A_1,\ldots,A_L\}$ then $\mathcal{C}_\alpha$ contains all choices $p_\alpha=\{p_{\alpha,t}:t>0\}$ which have, apart from property (2) in §2.2, the following restriction: if $y\in X(\alpha,P,<t)$, $x\in X(P_\alpha,\alpha,<t)$ and $s\in X(\alpha,P,t)$ then $p_{\alpha,t}(y,x,s)=0$ if $C(s)$ contains elements from $\mathfrak{T}$ which are not computable from the elements in $C(y)\cup C(x)$.

The formal credential mechanism can be described in a similar way as the "real" credential mechanism of §3; only all numbers in $\mathbf{Z}_N^*$ appearing in the real credential mechanism are replaced by their inverse images under $\psi$, where $\psi$ is some F-homomorphism. The formal

credential mechanism will satisfy the conditions of §3.5 with $\mathfrak{X}$ replacing $\mathbf{Z}_N^*$. The only exception is made for the elements of $\mathfrak{X}$ corresponding to the numbers $r_l, s_l$ chosen in step 3 of some $I(i_k, A_j)$: these will correspond to different elements of the set $\{H_1, \ldots, H_T\}$ which need not be chosen by means of a uniform distribution. We notice that in particular, $Z$ chooses the sets $\mathbb{S}$ in step 6 of each $I(i_k, A_j)$ uniformly from the subsets of $\{1, \ldots, n\}$ of cardinality $\frac{1}{2}n$ and independently of the other steps executed at the same moment or before.

Elements of $\mathfrak{X}$, chosen or computed in the formal credential mechanism will be denoted in the same way as the corresponding messages in the real credential mechanism of §3.3, except that lower case characters appearing in the bases of the expressions in §3.3 are replaced by corresponding capitals, that $f$ is replaced by $F$ and that exponents $\bar{b}$ are replaced by $b^{-1}$. Apart from that, steps in the formal credential mechanism will be denoted in the same way as the corresponding steps in §3.3. Subprotocols in the formal credential mechanism will be given the same names as the corresponding subprotocols in the real credential mechanism.

## 6. UNFORGEABILITY

In this section we formulate analogues of properties 1,2 and 3 mentioned in §1.1 for the formal credential mechanism, give a theorem, stating that the probability that these properties do not hold is bounded above by a number which is exponentially small in the security parameter $n$ appearing in step 3 of $I(i_k, A_j)$, and give an example of an attack, showing that the upper bound in the theorem is optimal.

### 6.1. Statement of the result

We shall use the same notation, and make the same assumptions, as in the previous sections. In particular, $\mathfrak{X}$ will have the meaning of §5.1, $\phi(N)$, $c_1, \ldots, c_K, a, p_1, \ldots, p_L, q_1, \ldots, q_L$ are pairwise coprime, and $p_1, \ldots, q_L$ are primes larger than $\frac{1}{2}n$. We put $b_j = p_j^2 c_1 \cdots c_K$ for $j = 1, \ldots, L$. For any $P \in \mathfrak{X}$ and $c \in \mathbf{Z}$ with $\gcd(c, \phi(N)) = 1$ we shall refer to $P^{c^{-1}}$ as "credential $c$ on pseudonym $P$". The same computational model for the formal module $\mathfrak{X}$ as introduced in §5.1 will be used.

Consider an attack $\mathcal{A}$ on the formal credential mechanism in which $Z$ does not cheat. Let $P$ be a pseudonym, used by some representative $g_j$ during an execution of $\mathcal{A}$. We say that $P$ is *properly validated* for organization $A_j$ during this execution if the following happens:
- in step 2 of $II(g_j, A_j)$, $g_j$ sends $P$ to $A_j$ together with some message $W_P$, and $P \neq Q$ and $PW_P \neq QW_Q$ for each other pseudonym $Q$ which was sent to $A_j$ together with $W_Q$ at the same moment or before in step 2 of some other subprotocol $II(g'_j, A_j)$;
- in step 5 of $II(g_j, A_j)$, $g_j$ sends $V_P, T_{2P}, \ldots, T_{\frac{1}{2}n,P}$ to $A_j$ such that

$$W_P = \prod_{l=2}^{\frac{1}{2}n} PT_{lP}^{b_l}, \quad V_P = P^{p_j^{-2}} \{F(P) \prod_{l=2}^{\frac{1}{2}n} F(PT_{lP}^{b_l})\}^{(p_j q_j)^{-1}}.$$

There is nothing which prevents organizations from accepting pseudonyms which are not properly validated. However, organizations accepting such pseudonyms will in the worst case only harm themselves, since $Z$ is only willing to compute credentials on pseudonyms which are properly validated. (cf. $II(g_j,A_j)$ and step 4 of $III(g_j,A_j,c)$ in §3.3).

We say that a set of pseudonyms, properly validated during an execution of $\mathcal{A}$, has the *unforgeability property* if it satisfies the following three analogues of properties 1,2 and 3 of §1.1:

- it can be partitioned in two ways:
  in I-sets each containing the pseudonyms used by the representatives of a fixed individual, and O-sets each containing the pseudonyms which have been properly validated for a fixed organization;
- each I-set and each O-set have at most one pseudonym in common;
- if $P$ is a pseudonym, properly validated before moment $t$, and $c$ a credential, then $P^{c^{-1}}$ is computable from $\mathcal{K}$ and the set of elements of $\mathcal{Z}$ received by the individuals before moment $t$ if and only if this set contains $Q^{c^{-1}}$ where $Q$ is a pseudonym belonging to the same I-set as $P$.

The set of pseudonyms for the attack $\mathcal{A}$ is defined as the stochastic variable of which each value is the set of pseudonyms properly validated during an execution of $\mathcal{A}$. Henceforth we shall implicitly assume, when speaking of an attack on the formal credential mechanism, that the choice of any participant $\alpha$ in this attack belongs to the collection $\mathcal{C}_\alpha$ described in §5.3. The following theorem is proved in §7.

**THEOREM 2.** *Let $\mathcal{A}$ be any attack on the formal credential mechanism in which $Z$ does not cheat (possibly the credential mechanism itself). Then the probability that the set of pseudonyms for $\mathcal{A}$ does not have the unforgeability property, is at most $LR\times(\binom{n}{\frac{1}{2}n})^{-1}$.*

**Remark 1.** By Stirling's formula, the upper bound mentioned in Theorem 2 is approximately equal to $LR\times(\frac{1}{2}\pi n)^{\frac{1}{2}}2^{-n}$. In the next subsection we shall describe an attack, showing that the upper bound for the probability in Theorem 2 can not be essentially improved.

**Remark 2.** Each attack on the formal credential mechanism can be "translated" into an attack on the real credential mechanism of §3 by means of an F-homomorphism $\psi$. In fact, each attack on the real credential mechanism can be considered as such a translation, if values of $\mathbf{Z}_N^*$, obtained during an execution of such an attack by other means than multiplications, multiplicative inversions or applications of $f$, are assigned to $\psi(H_1), \ldots, \psi(H_T)$.

As a consequence of the remark made at the end of §5.2 it is still unknown whether there is an attack on the real credential mechanism which gives individuals a chance considerably larger than $LR\times(\binom{n}{\frac{1}{2}n})^{-1}$ of getting validators for a set of pseudonyms not having the properties 1,2 or 3 mentioned in §1.1.

## 6.2. Attacks on the formal credential mechanism

It will be proved in §7, that a set of pseudonyms, constructed as prescribed in the formal credential mechanism, will have the unforgeability property. Therefore, individuals might only obtain a set of pseudonyms without the unforgeability property by means of some attack on the formal credential mechanism. We assume that $Z$ is trusted by the organizations, whence that $Z$ does not cheat in such an attack. The choices of the individuals in such an attack will have the constraints described in §5.3.

Cheating individuals will only be able to compute a set of validators for a set of pseudonyms without the unforgeability property if they succeed, by means of some attack, in getting elements of $\mathfrak{L}$ from which these validators can be computed. There are two ways by which cheating individuals could get such elements. The first way is a kind of cooperation, in which each cheating individual sends all his received elements of $\mathfrak{L}$ to the other cheating individuals. Such a cooperation can even take place with organizations. The second method by which a cheating individual $i_k$ may obtain appropriate elements of $\mathfrak{L}$ is by sending, in step 5 of some $I(i_k,A_j)$, elements $A_{kl}$ $(l=1,\ldots,n)$ to $Z$ which are not all computed in the way described in step 4. Since the set $\mathbb{S}$, generated by $Z$ in step 6, will be chosen uniformly from all subsets of $\{1,\ldots,n\}$ of cardinality $\frac{1}{2}n$, and independently of all what happened previously in the attack, $i_k$ may have a considerable chance of not being able to show all $R_l,S_l$ to $Z$ in step 8. Moreover, even if $i_k$ is able to show all these $R_l,S_l$, the probability that he will get a validator $V_{kj}$ in step 11 which is useful for his purposes, is in general quite small.

**An example of an attack.** We shall describe an attack on the formal credential mechanism in which a single individual tries to obtain a validator for a pseudonym on which he can compute each credential he likes just by himself. In this attack he need not cooperate with other participants of the credential mechanism.

Let $i_k,A_j$ be an individual and an organization and let $g_j$ be $i_k$'s representative communicating with $A_j$. Let $a, p_j, q_j, c_1, \ldots, c_K$ have the same meaning as in §3.3, and put $p=p_j$, $q=q_j$, $c=c_1\cdots c_K$, $b=p^2c$. Since $\gcd(p,c)=1$ there are integers $\alpha,\beta$ such that $\alpha p^2+\beta c=1$, which $i_k$ can compute easily by means of Euclid's algorithm. All steps we refer to will be in $I(i_k,A_j)$. In step 3 $i_k$ chooses $2n$ elements $R_l,S_l$ $(l=1,\ldots,n)$ of $\mathfrak{L}$, as in the formal credential mechanism. In step 4 he computes

$$A_{kl}:=F(U_k^{1-\alpha p^2}(M_lR_l^a)^b)S_l^{pq} \text{ for } l=1,\ldots,\tfrac{1}{2}n,$$
$$A_{kl}:=F(U_k(M_lR_l^a)^b)S_l^{pq} \text{ for } l=\tfrac{1}{2}n+1,\ldots,n.$$

$i_k$ is able to show $R_l,S_l$ $(l\in\mathbb{S})$ to $Z$ in step 8 which for all $l$ in $\mathbb{S}$ satisfy the condition of step 9, if and only if $Z$ chooses $\mathbb{S}=\{\frac{1}{2}n+1,\ldots,n\}$ in step 6. The probability that $Z$ chooses this set is $\binom{n}{\frac{1}{2}n}^{-1}$. Provided that the check in step 9 gives the value 'true', $Z$ sends to $i_k$ the validator

$$V_{kj}:=U_k^{p^{-2}}\{\prod_{l=1}^{\frac{1}{2}n}A_{kl}\}^{(pq)^{-1}}.$$

By multiplying $V_{kj}$ with $U_k^{-\alpha}(M_1R_1^a)^b\prod_{l=1}^{\frac{1}{2}n}S_l^{-1}$, and using that $1-\alpha p^2=\beta c$, $i_k$ obtains

$$\tilde{V}_{g_j}:=U_{g_j}^{p^{-2}}\{\prod_{l=1}^{\frac{1}{2}n}F(U_{g_j}T_l^b)\}^{(pq)^{-1}},$$

where

$$U_{g_j}=U_k^{1-\alpha p^2}(M_1R_1^a)^b = U_k^{\beta c}(M_1R_1^a)^b,$$

$T_1=1$, and $T_l=M_lR_l^a(M_1R_1^a)^{-1}$ for $l=1,\ldots,\frac{1}{2}n$.

When $g_j$ sends $U_{g_j}$, $\tilde{W}_{g_j}:=\prod_{l=2}^{\frac{1}{2}n}U_{g_j}T_l^b$, $\tilde{V}_{g_j}$, and $T_l$ ($l=2,\ldots,\frac{1}{2}n$) to $A_j$ during $II(g_j,A_j)$, then $A_j$ will accept $U_{g_j}$ as a pseudonym. $i_k$ is now able to compute $U_{g_j}^{c_t^{-1}}$ for each credential $c_t$ and show these to $A_j$ even when no other organization has issued this credential to one of $i_k$'s representatives. Hence no set of pseudonyms containing $U_{g_j}$ can have the unforgeability property. We repeat that the probability of success for $i_k$ is at most $(\frac{n}{\frac{1}{2}n})^{-1}$. If all $R$ individuals would try this attack for all $L$ organizations, and if $LR$ is small compared with $(\frac{n}{\frac{1}{2}n})$, then the chance that at least one of the individuals is successful is approximately $LR\times(\frac{n}{\frac{1}{2}n})^{-1}$. This shows that Theorem 2 is optimal.

## 7. PROOF OF THEOREM 2

In §7.1 we introduce some notation, needed in the proof of Theorem 2. In §7.2 we prove that a set of pseudonyms, constructed in the way prescribed in the formal credential mechanism, has the unforgeability property, and in §§7.3-7.5 we shall prove that whatever attack they try, individuals will have only a very small chance of being able to validate properly a pseudonym which is not of the form described in the formal credential mechanism.

### 7.1. Notation

For any integral domain $\mathfrak{R}$ with unity, we denote by $\mathfrak{R}^t$ ($t=1,2,\cdots$) the $\mathfrak{R}$-module of $t$-tuples $(x_1,\ldots,x_t)$ over $\mathfrak{R}$, and by $\mathfrak{R}^\infty$ the $\mathfrak{R}$-module of infinite tuples $(x_1,x_2,\cdots)$ over $\mathfrak{R}$ of which at most finitely entries are non-zero. The tuple of which all entries are 0, is in each module $\mathfrak{R}^t$ ($t=\infty,1,2,...$) denoted by $\mathbf{0}$.

Let $\mathbb{N},\mathbb{Z},\mathbb{Q}$ denote the set of positive integers, the set of all integers, and the set of rational numbers, respectively. For any $c\in\mathbb{N}$, put

$$\tilde{\mathbb{Q}}_c = \{\frac{a}{b} : a,b\in\mathbb{Z},\gcd(b,c\phi(N))=1\}.$$

In particular, $\tilde{\mathbb{Q}}=\tilde{\mathbb{Q}}_1$. If $\alpha,\beta\in\tilde{\mathbb{Q}}$ we write $\alpha\equiv\beta \bmod c$ if there is a $\gamma\in\tilde{\mathbb{Q}}_c$ with $\alpha-\beta=\gamma c$. If $\mathbf{a}=(a_1,a_2,\cdots)$, $\mathbf{b}=(b_1,b_2,\cdots)\in\tilde{\mathbb{Q}}^t$ ($t\in\{\infty,1,2,\cdots\}$) then we write $\mathbf{a}\equiv\mathbf{b} \bmod c$ if $a_i\equiv b_i$

mod $c$ for $i=1,2,\cdots$.

We shall use the same notation as in the previous sections. In particular, we have $\mathfrak{L}=\tilde{\mathbf{Q}}[F,\mathfrak{K}]=\tilde{\mathbf{Q}}[\mathfrak{K}\cup\mathfrak{F}]$, where, similar to §5.1, $\mathfrak{K}=\{U_1,\ldots,U_R,M_1,\ldots,M_n,H_1,\ldots,H_T\}$ and $\mathfrak{F}$ is the set of images of $F$. Since $\mathfrak{F}$ is enumerable we may write $\mathfrak{F}=\{F_1,F_2,F_3,\cdots\}$. Thus each element of $\mathfrak{L}$ can be written as

$$\prod_{i=1}^{R}U_i^{x_i}\prod_{i=1}^{\infty}F_i^{y_i}\prod_{i=1}^{n}M_i^{w_i}\prod_{i=1}^{T}H_i^{z_i} \tag{19}$$

or, more compactly, as

$$U^{\mathbf{x}}F^{\mathbf{y}}M^{\mathbf{w}}H^{\mathbf{z}},$$

where

$$\mathbf{x}=(x_1,\ldots,x_R)\in\tilde{\mathbf{Q}}^R,\ \mathbf{y}=(y_1,y_2,\cdots)\in\tilde{\mathbf{Q}}^\infty,$$
$$\mathbf{w}=(w_1,\ldots,w_n)\in\tilde{\mathbf{Q}}^n,\ \mathbf{z}=(z_1,\ldots,z_T)\in\tilde{\mathbf{Q}}^T,$$

and $U^{\mathbf{x}}$, $F^{\mathbf{y}}$, $M^{\mathbf{w}}$ and $H^{\mathbf{z}}$ are abbreviations of

$$\prod_{i=1}^{R}U_i^{x_i},\ \prod_{i=1}^{\infty}F_i^{y_i},\ \prod_{i=1}^{n}M_i^{w_i},\ \text{and}\ \prod_{i=1}^{T}H_i^{z_i},$$

respectively.

Similar to the previous sections, $Z$ denotes the signature authority, $A_1,\ldots,A_L$ the organizations, and $i_1,\ldots,i_R$ the individuals participating in the formal credential mechanism. As before, $c_1,\ldots,c_K$, $a$, $p_1,\ldots,p_L$, $q_1,\ldots,q_L$ are positive integers which are pairwise coprime, and coprime with $\phi(N)$, and $p_1,\ldots,q_L$ are primes larger than $\frac{1}{2}n$. From now on we assume, when referring to the formal credential mechanism, that it is modified in the following way: whenever $Z$ or some representative must send an element of $\mathfrak{L}$ to some individual, then it sends it to all individuals. Thus all individuals will have exactly the same computational abilities in $\mathfrak{L}$. It is obviously sufficient to prove Theorem 2 for this modified formal credential mechanism.

We say that $X\in\mathfrak{L}$ is *computable by the individuals*, or that the individuals can compute $X$, at moment $t$ if $X$ is computable from $\mathfrak{K}$ and the elements of $\mathfrak{L}$ received by the individuals before moment $t$ during an execution of an attack on the formal credential mechanism. By saying that the individuals can compute $X$ before moment $\infty$ we mean that there is a moment at which the individuals can compute $X$. We shall need the following obvious but important fact: if $\mathfrak{W}=\{W_1,\ldots,W_S\}$ is a subset of $\mathfrak{L}$ containing $\mathfrak{K}$, then each element of $\mathfrak{L}$ which is computable from $\mathfrak{W}$ can be written in the form

$$\prod_{i=1}^{S}W_i^{n_i}\prod_{i=1}^{\infty}F_i^{y_i} \tag{20}$$

where the $n_i$ and $y_i$ are all integers such that at most finitely many of the $y_i$ are non-zero, and

$y_i=0$ if $F_i=F(X)$ and $X$ is not computable from $\mathfrak{W}$.

### 7.2. Proof of unforgeability if pseudonyms are properly formed

If $P$ is a pseudonym, used by some representative $g_j$, which has been properly validated for some organization $A_j$, then $t_P$ denotes the moment at which the validation of this pseudonym was completed, that is the moment at which step 5 of $\Pi(g_j,A_j)$ was executed.

Consider an execution of some attack in the formal credential mechanism and suppose that the set of pseudonyms properly validated during this attack, $\mathfrak{S}$ say, has the following properties:

- for each $P\in\mathfrak{S}$, there are a pair $(k,j)$ with $1\leqslant k\leqslant R, 1\leqslant j\leqslant L$, and $R_{kj}\in\mathfrak{L}$ such that

$$P=U_kR_{kj}^{b_j} \text{ and the individuals can compute } R_{kj} \text{ at moment } t_P; \tag{21}$$

- for each pair $(k,j)$ with $1\leqslant k\leqslant R, 1\leqslant j\leqslant L$ there is at most one pseudonym $P$ in $\mathfrak{S}$ satisfying (21).

Then we have

**Lemma 3.** *$\mathfrak{S}$ has the unforgeability property.*

**Proof.** Define the I-sets $I_1,\ldots,I_R$ and O-sets $O_1,\ldots,O_L$ such that the pseudonym in $\mathfrak{S}$ which satisfies (21) belongs to $I_k$ and $O_j$. Then each I-set and each O-set have at most one pseudonym in common. Let $c$ be a credential, $P$ a pseudonym in $I_1$ which has been validated before moment $t$, and suppose that the set consisting of $\mathfrak{K}$ and the elements of $\mathfrak{L}$ received by the individuals before moment $t$, does not contain any $Q^{c^{-1}}$ with $Q\in I_1$. We shall prove that the individuals can not compute $P^{c^{-1}}$ at moment $t$. By repeating the same argument for the other I-sets, one proves that $\mathfrak{S}$ has the unforgeability property.

Let

$$\mathfrak{g}=\{U^{\mathbf{x}}F^{\mathbf{y}}M^{\mathbf{w}}H^{\mathbf{z}}:\mathbf{x}=(x_1,\ldots,x_R)\in\tilde{\mathbf{Q}}^R,x_1\in\tilde{\mathbf{Q}}_c,\mathbf{y}\in\tilde{\mathbf{Q}}^{\infty},\mathbf{w}\in\tilde{\mathbf{Q}}^n,\mathbf{z}\in\tilde{\mathbf{Q}}^T\}$$

and $\mathfrak{D}(u)$ the set of elements of $\mathfrak{L}$ computable by the individuals at moment $u$ for each moment $u$. We shall prove that $\mathfrak{D}(t)\subset\mathfrak{g}$. Obviously, $\mathfrak{D}(0)=\mathbf{Z}[F,\mathfrak{K}]$ is contained in $\mathfrak{g}$. For each $u$ with $1\leqslant u\leqslant t$ we show that $\mathfrak{D}(u-1)\subset\mathfrak{g}$ implies $\mathfrak{D}(u)\subset\mathfrak{g}$. Then it follows by induction on $u$ that $\mathfrak{D}(t)\subset\mathfrak{g}$.

Fix $u$ with $1\leqslant u\leqslant t$. To establish our induction step, we have to consider the set of elements of $\mathfrak{L}$ received by the individuals at moment $u-1$. This set consists of either credentials on pseudonyms or validators. Suppose that some individual receives a validator at moment $u-1$. This validator is a $p_j^2q_j$-th root on some element of $\mathfrak{D}(u-2)$. Since $\mathfrak{g}$ is closed under exponentiation with numbers in $\tilde{\mathbf{Q}}_c$, this shows that this validator belongs to $\mathfrak{g}$. By a similar argument it follows that credentials $\neq c$ on pseudonyms received by some individual at moment $u$ must belong to $\mathfrak{g}$. Suppose that at moment $u-1$ some credential $Q^{c^{-1}}$ on a pseudonym $Q$ has been received by some individual. Then by assumption, $Q$ must belong to one of the sets $I_k$ with

$k>1$. Moreover, $Q$ must have been properly validated before moment $u-1$. Hence by (21), $Q$ can be written as $U_k R_{kj}^{b_j}$ for some $k>1$ and $j$ and some $R_{kj}$ in $\mathfrak{D}(u-2)$. Since $c$ divides $b_j$, this shows that $Q^{c^{-1}} \in \mathcal{G}$. We infer that all elements of $\mathfrak{L}$, received by the individuals at moment $u-1$ belong to $\mathcal{G}$. But since $\mathcal{G}$ is closed under computations, this proves that $\mathfrak{D}(u)$ is contained in $\mathcal{G}$.

We shall now show that $P^{c^{-1}} \notin \mathcal{G}$. Assume the contrary. By (21) and the fact that $P$ has been properly validated before moment $t$, there are $j$ and $R_{1j} \in \mathfrak{D}(t)$ such that $P = U_1 R_{1j}^{b_j}$. Since $c$ divides $b_j$ this implies that $U_1^{c^{-1}}$ belongs to $\mathfrak{D}(t)$, whence to $\mathcal{G}$, which is false. Hence our assumption must have been wrong. This completes the proof of Lemma 3. □

Roughly speaking, Lemma 3 proves that any set of pseudonyms having the form prescribed in the formal credential mechanism, must have the unforgeability property. In order to prove Theorem 2, it is therefore sufficient to show that the probability that individuals are able to validate properly pseudonyms which are not of the required form is very small. We shall state this more precisely.

By an *$A_j$-validator* for a pseudonym $P$ we shall mean a tuple

$$V_j(P) = (V_P, W_P, T_{2P}, \ldots, T_{½n,P}),$$

of elements in $\mathfrak{L}$ such that

$$V_P = P^{p_j^{-2}} \left( F(P) \prod_{l=2}^{½n} F(P T_{lP}^{b_j}) \right)^{(p_j q_j)^{-1}}, \quad W_P = \prod_{l=2}^{½n} P T_l^{b_j}, \tag{22}$$

where $p_j, q_j, b_j$ have the same meaning as before. If $P$ has been properly validated for $A_j$ then $A_j$ has received an $A_j$-validator for $P$. Two $A_j$-validators $V_j(P_1) = (V_{P_1}, W_{P_1}, T_{2,P_1}, \ldots, T_{½n,P_1})$ and $V_j(P_2) = (V_{P_2}, W_{P_2}, T_{2,P_2}, \ldots, T_{½n,P_2})$ for pseudonyms $P_1$ and $P_2$, respectively, are called *equivalent* if, after reordering, the tuples $(P_1, P_1 T_{2,P_1}^{b_j}, \ldots, P_1 T_{½n,P_1}^{b_j})$ and $(P_2, P_2 T_{2,P_2}^{b_j}, \ldots, P_2 T_{½n,P_2}^{b_j})$ are equal. (Loosely speaking this means that that the products of the $F$-values appearing in $V_{P_1}$ and $V_{P_2}$, respectively, are equal). Note that in none of these tuples, the entries need be distinct. If the validators $V_j(P_1)$ and $V_j(P_2)$ are equivalent, then $P_1 W_{P_1} = P_2 W_{P_2}$. Hence any two pseudonyms which have been properly validated for $A_j$ must have been shown to $A_j$ together with inequivalent validators.

For each attack on the formal credential mechanism, and each $j$ in $\{1, \ldots, L\}$, the following events are defined:

$\boldsymbol{E_{1j}}$: in some execution, there is a pseudonym $P$ in $\mathfrak{L}$, properly validated for $A_j$, such that none of the roots $(P U_k^{-1})^{b_j^{-1}}$ can be computed by the individuals at moment $t_P$, for $k = 1, \ldots, R$.

$\boldsymbol{E_{2j}}$: in some execution, there is a moment $t$ at which the individuals can compute: two pseudonyms $P_1$ and $P_2$; the $b_j$-th root of their quotient, $(P_1 P_2^{-1})^{b_j^{-1}}$; and two *inequivalent* $A_j$-validators $V_j(P_1)$ and $V_j(P_2)$ for $P_1$ and $P_2$ respectively.

Informally, $E_{1j}$ is the event, that a pseudonym, properly validated for $A_j$, is not of the form

prescribed in the credential mechanism, and $E_{2j}$ is the event that two "similar" pseudonyms have been properly validated with "non-similar" validators.

Suppose that in some execution of an attack on the formal credential mechanism, none of the events $E_{1j}$, $E_{2j}$ $(1 \leqslant j \leqslant L)$ takes place. Let $\mathfrak{S}$ be the set of properly validated pseudonyms in this execution. Then each pseudonym in $\mathfrak{S}$ must satisfy (21) for some pair $(k,j)$, since none of the events $E_{1j}$ takes place. Moreover, no two pseudonyms in $\mathfrak{S}$ can satisfy (21) for the same pair $(k,j)$. For if two pseudonyms, $P_1$, $P_2$ say, in $\mathfrak{S}$, would have satisfied (21) for the same pair $(k,j)$, then at some moment $t$ the individuals would have been able to compute $P_1$, $P_2$ and $(P_1 P_2^{-1})^{b_j^{-1}}$. But since none of the events $E_{2j}$ was supposed to take place, the validators for $P_1$ and $P_2$ must be equivalent, which contradicts the fact that these pseudonyms have been properly validated.

Together with Lemma 3 this implies the following: the probability that the set of pseudonyms for (an attack on) the formal credential mechanism does not have the unforgeability property is at most equal to the probability of event $E_{11} \cup E_{12} \cup \cdots \cup E_{1L} \cup E_{2L}$. Theorem 3 states, in a more precise form, that this event has probability at most $LR \times (\frac{n}{n/2})^{-1}$. This implies Theorem 2 at once.

**THEOREM 3.** *For each attack on the formal credential mechanism in which Z does not cheat, we have*

$$Pr[E_{1j}] \leqslant R \times (\tfrac{n}{n/2})^{-1} \text{ and } Pr[E_{2j}] = 0 \text{ for } j = 1, \ldots, L.$$

### 7.3. Preliminaries to the proof of Theorem 3

We shall use the same notation as in the previous sections. In particular, $i_1, \ldots, i_R$ will be the individuals participating in the credential mechanism. We fix $j$ in $\{1, \ldots, L\}$ and put $p = p_j$, $q = q_j$ and $b = b_j$. As mentioned before, each element of $\mathfrak{X}$ can be expressed as a finite product of powers of which the bases belong to $\mathfrak{K} \cup \mathfrak{F}$ and the exponents to $\tilde{\mathbf{Q}}$. We shall show, that any condition that the individuals can ever compute certain elements of $\mathfrak{X}$ can be expressed as the solvability of some system of linear equations modulo $p^2 q$ of which some of the coefficients are stochastic variables and the unknowns belong to $\tilde{\mathbf{Q}}$.

Consider an attack on the formal credential mechanism, and denote the changed version of $I(i_k, A_j)$ in this attack also by $I(i_k, A_j)$. In step 5 of $I(i_k, A_j)$, $i_k$ sends elements $A_{kl}$ $(l = 1, \ldots, n)$ to $Z$, which might have been computed in an other way than described in the formal credential mechanism, and might have been chosen by means of a probability distribution depending on all steps previously executed in which some individual was involved. In step 6, $Z$ chooses a set $\mathfrak{S}_k$ (which is now indexed in order to distinguish sets $\mathfrak{S}$ generated in different $I(i_k, A_j)$), by means of a probability distribution which is uniform on the collection of subsets of $\{1, \ldots, n\}$ of cardinality $\frac{1}{2}n$, and independent of all other steps executed at the same moment or before in the credential mechanism. In step 8, $i_k$ has to send $R_l, S_l$ to $Z$ for each $l$ in $\mathfrak{S}_k$ such that

$A_{kl}=F(U_k(M_lR_l^a)^b)S_l^{pq}$. If $Z$ concludes in step 9 that indeed $i_k$ constructed each $A_{kl}$ with $l\in\mathcal{S}_k$ in the proper way, then he sends in step 11,

$$V_{kj}=U_k^{p^{-2}}\{\prod_{l\in\mathcal{S}_k}A_{kl}\}^{(pq)^{-1}}$$

to $i_k$.

From now on, we exclude attacks useless to the individuals in which some security check by $Z$ in $I(i_k,A_j)$ other than that in step 9 is 'false', or that $i_k$, while knowing the correct $R_l,S_l$ for some $l$ in $\mathcal{S}_k$ in step 8, sends false $R_l$ or $S_l$ to $Z$.

For each $k$ in $\{1,\ldots,R\}$ we introduce the following notation:

$$Y_k=\prod_{l=1}^{n}A_{kl}\ ,\quad V_k=\left[U_k^{p^{-2}}(Y_k\{\prod_{l\in\mathcal{S}_k}F(U_k(M_lR_l^a)^b)\}^{-1})^{(pq)^{-1}}\right]^{\xi_k}\ ,$$

where $\xi_k=1$ if the check in step 9 of $I(i_k,A_j)$ gave the value 'true' and $\xi_k=0$ otherwise. Hence if the check in step 9 did not fail, $V_k$ is equal to $V_{kj}\prod_{l\in\mathcal{S}_k}S_l$, whereas $V_{kj}=1$ if this check failed. We notice, that $Y_k$ is controlled completely by $i_k$; it is independent of $\mathcal{S}_k$. We assume from now on that $i_k$ receives $V_k$ from $Z$ instead of $V_{kj}$ in step 11, for $k=1,\ldots,R$. This does not change the computational abilities of the individuals.

Let $t_k$ be the moment at which step 11 of $I(i_k,A_j)$ is executed, that is the moment at which $i_k$ (and so the other individuals) receives $V_k$ from $Z$. We shall derive the upper bounds in Theorem 3 subject to the condition that

$$t_1\leqslant t_2\leqslant\cdots\leqslant t_R. \tag{23}$$

By repeating the argument for each other possible order of the $t_k$'s, one can prove Theorem 3.

Henceforth we shall assume (23). Each $i_k$ sends $A_{k1},\ldots,A_{kn}$ to $\mathfrak{L}$ before moment $t_k$. Hence for $k=1,\ldots,R$, the individuals can compute $Y_k$ before they receive $V_k,\ldots,V_R$. In view of (20), the $Y_k$'s have the form

$$\begin{aligned}&Y_1=U^{\mathbf{x}_1}F^{\mathbf{y}_1}M^{\mathbf{w}_1}H^{\mathbf{z}_1}\ ,\\&Y_k=V_1^{\delta_{1k}}\cdots V_{k-1}^{\delta_{k-1,k}}U^{\mathbf{x}_k}F^{\mathbf{y}_k}M^{\mathbf{w}_k}H^{\mathbf{z}_k}\text{ for }k=2,\ldots,R\ ,\end{aligned} \tag{24}$$

where $\delta_{1k},\ldots,\delta_{k-1,k}\in\tilde{\mathbf{Q}}_{pq}$, $\mathbf{x}_k\in\tilde{\mathbf{Q}}_{pq}^R$, $\mathbf{y}_k\in\tilde{\mathbf{Q}}_{pq}^\infty$, $\mathbf{w}_k\in\tilde{\mathbf{Q}}_{pq}^n$, $\mathbf{z}_k\in\tilde{\mathbf{Q}}_{pq}^T$. Apart from $V_1,\ldots,V_R$, individuals may receive validators for other organizations, or credentials on pseudonyms, which are all $d$-th roots on messages previously computable by the individuals, where $d$ is a positive integer coprime with $pq$. We took this into consideration by allowing that the coefficients in (24) belong to $\tilde{\mathbf{Q}}_{pq}$ rather than $\mathbf{Z}$.

For $k=1,\ldots,R$, let $\mathfrak{T}_k$ be the set of positive integers $j$ such that $F_j=F(U_k(M_lR_l^a)^b)$ for some $R_l$ sent by $i_k$ to $Z$ in step 8 of $I(i_k,A_j)$, and define $\mathbf{e}(\mathfrak{T}_k)\in\tilde{\mathbf{Q}}^\infty$ by

$$F^{\mathbf{e}(\mathfrak{T}_k)} = \prod_{l \in \mathfrak{S}_k} F(U_k(M_l R_l^q)^p) .$$

Moreover, $\mathbf{d}_k \in \tilde{\mathbf{Q}}_R$ denotes the vector of which the $k$-th coordinate is 1, and the other coordinates are 0. With this notation we have

$$V_k = \left( U^{p^{-2}\mathbf{d}_k} \left( Y_k F^{-\mathbf{e}(\mathfrak{T}_k)} \right)^{(pq)^{-1}} \right)^{\xi_k} \text{ for } k = 2, \ldots, R . \tag{25}$$

The following lemma is crucial in the proof of Theorem 3.

**Lemma 4.** *Consider an execution of some attack on the formal credential mechanism in which $Y_1, \ldots, Y_R$ satisfy* (24) *and $V_1, \ldots, V_R$ satisfy* (25). *Suppose that at a moment $< t_{h+1}$ (where $h \in \{1, \ldots, R\}$ and $t_{R+1} := \infty$), the individuals can compute $U^{\mathbf{u}} F^{\mathbf{f}} M^{\mathbf{m}} H^{\mathbf{h}} \in \mathfrak{L}$, where $\mathbf{u} \in \tilde{\mathbf{Q}}^R$, $\mathbf{f} \in \tilde{\mathbf{Q}}^\infty$, $\mathbf{m} \in \tilde{\mathbf{Q}}^n$ and $\mathbf{h} \in \tilde{\mathbf{Q}}^T$. Then there are $m_1, \ldots, m_h \in \tilde{\mathbf{Q}}$ such that*

$$\sum_{k=1}^{h} m_k \xi_k (\mathbf{d}_k + pq^{-1}\mathbf{x}_k) \equiv p^2 \mathbf{u} \bmod p^2 , \tag{26}$$

$$\sum_{k=1}^{h} m_k \xi_k (\mathbf{y}_k - \mathbf{e}(\mathfrak{T}_k)) \equiv pq\mathbf{f} \bmod p , \tag{27}$$

$$\sum_{k=1}^{h} m_k \xi_k (\mathbf{y}_k - \mathbf{e}(\mathfrak{T}_k)) \equiv pq\mathbf{f} \bmod q . \tag{28}$$

*Moreover, if $p^2\mathbf{u} \in \tilde{\mathbf{Q}}_p^R$, then $m_1\xi_1, \ldots, m_h\xi_h$ belong to $\tilde{\mathbf{Q}}_p$.*

**Proof.** By (21) there are $n_1, \ldots, n_h \in \tilde{\mathbf{Q}}_{pq}$, $\mathbf{a} \in \tilde{\mathbf{Q}}_{pq}^R$, $\mathbf{b} \in \tilde{\mathbf{Q}}_{pq}^\infty$, $\mathbf{c} \in \tilde{\mathbf{Q}}_{pq}^n$ and $\mathbf{d} \in \tilde{\mathbf{Q}}_{pq}^T$ such that

$$U^{\mathbf{u}} F^{\mathbf{f}} M^{\mathbf{m}} H^{\mathbf{h}} = V_1^{n_1} \cdots V_h^{n_h} U^{\mathbf{a}} F^{\mathbf{b}} M^{\mathbf{c}} H^{\mathbf{d}} . \tag{29}$$

For $k = 1, \ldots, R$, put

$$W_k = U_k^{p^{-2}} \left( U^{\mathbf{x}_k} F^{\mathbf{y}_k - \mathbf{e}(\mathfrak{T}_k)} M^{\mathbf{w}_k} H^{\mathbf{z}_k} \right)^{(pq)^{-1}} . \tag{30}$$

Then by (24) and (25),

$$W_k^{\xi_k} = V_k \left( \prod_{j=1}^{k-1} V_j^{-\delta_{jk}\xi_k} \right)^{(pq)^{-1}} .$$

Using these relationships, it is possible to express each $V_k$ as a product of powers of $W_1^{\xi_1}, \ldots, W_k^{\xi_k}$, in which the exponents belong to $\tilde{\mathbf{Q}}$ and may have denominators divisible by $p$ or $q$. Together with (29) this shows that there are $m_1, \ldots, m_h$ in $\tilde{\mathbf{Q}}$ such that

$$U^{\mathbf{u}} F^{\mathbf{f}} M^{\mathbf{m}} H^{\mathbf{h}} = W_1^{m_1\xi_1} \cdots W_h^{m_h\xi_h} U^{\mathbf{a}} F^{\mathbf{b}} M^{\mathbf{c}} H^{\mathbf{d}} .$$

By combining this with (30) and equating the exponents of $U$ and $F$, we obtain

$$\sum_{k=1}^{h} m_k\xi_k(p^{-2}\mathbf{d}_k+(pq)^{-1}\mathbf{x}_k)+\mathbf{a}=\mathbf{u},$$

$$\sum_{k=1}^{h} m_k\xi_k(pq)^{-1}(\mathbf{y}_k-\mathbf{e}(\mathcal{T}_k))+\mathbf{b}=\mathbf{f}.$$

(We shall not need the relationships coming from the exponents of $M$ and $H$). Now we obtain (26) by multiplying the first equation with $p^2$ and reducing it modulo $p^2$, and (27), (28) by multiplying the second equation with $pq$ and reducing it modulo $p$ and $q$, respectively.

Suppose that $p^2\mathbf{u}\in\tilde{\mathbf{Q}}_p^R$, and that not all numbers $\xi_k m_k$ with $1\leqslant k\leqslant h$ belong to $\tilde{\mathbf{Q}}_p$. Then there is an integer $d$, divisible by $p$, such that all numbers $dm_k\xi_k$ are integers and at least one of them is not divisible by $p$. But by multiplying (26) with $d$ and reducing it modulo $p$, we obtain

$$\sum_{k=1}^{h} dm_k\xi_k\mathbf{d}_k\equiv 0 \bmod p,$$

whence $dm_k\xi_k\equiv 0 \bmod p$ for $k=1,\ldots,h$. This contradiction shows that all $m_k\xi_k$ must belong to $\tilde{\mathbf{Q}}_p$ if $p^2\mathbf{u}\in\tilde{\mathbf{Q}}_p^R$. This completes the proof of Lemma 4. □

The following consequence of Lemma 4 will be useful.

**Lemma 5.** *Let* $\mathbf{f}\in\mathbf{Z}^\infty$ *and suppose that all coordinates of* $\mathbf{f}$ *have absolute values smaller than* $p$. *If there is a moment at which the individuals can compute* $F^{(pq)^{-1}\mathbf{f}}$, *then* $\mathbf{f}=\mathbf{0}$.

**Proof.** Suppose that at some moment, the individuals can compute $F^{(pq)^{-1}\mathbf{f}}$. Then by Lemma 4, eq. (26), there are $m_1,\ldots,m_R$, with $m_1\xi_1,\ldots,m_R\xi_R\in\tilde{\mathbf{Q}}_p$, such that

$$\sum_{k=1}^{R} m_k\xi_k(\mathbf{d}_k+pq^{-1}\mathbf{x}_k)\equiv\mathbf{0} \bmod p^2.$$

By reducing this equation modulo $p$, and using that $q$ is coprime with $p$, we obtain $m_k\xi_k\equiv 0 \bmod p$ for $k=1,\ldots,R$. A substitution of this into (27) yields that $\mathbf{f}\equiv\mathbf{0} \bmod p$. But since by assumption, the absolute values of the coordinates of $\mathbf{f}$ are smaller than $p$ this proves Lemma 5. □

### 7.4. Proof of $Pr[E_{2j}]=0$

Consider an execution of some attack on the formal credential mechanism in which $Z$ does not cheat. Suppose that in this execution there is a moment $t$ at which the individuals can compute pseudonyms $P_1,P_2$, the $b$-th root of their quotient $(P_1P_2^{-1})^{b^{-1}}$ and $A_j$-validators $V_j(P_1),V_j(P_2)$ for $P_1$ and $P_2$ respectively, where

$$V_j(P_i)=(V_{P_i},W_{P_i},T_{2,P_i},\ldots,T_{\frac{1}{2}n,P_i}) \text{ for } i=1,2.$$

We have to prove that $V_j(P_1)$ and $V_j(P_2)$ are equivalent.

Put $T_{1,P_1}=T_{1,P_2}=1$ and let $\mathbf{f}(i)\in\tilde{\mathbf{Q}}^\infty$ be defined by

$$F^{\mathbf{f}(i)} = \prod_{l=1}^{\frac{1}{2}n} F(P_i T_{l,P_i}^{b})$$

for $i = 1,2$. Then

$$V_{P_i} = P_i^{p^{-2}} F^{(pq)^{-1}\mathbf{f}(i)}$$

for $i = 1,2$. At moment $t$ the individuals can compute

$$(P_2 P_1^{-1})^{b^{-1}} V_{P_1} V_{P_2}^{-1} = F^{(pq)^{-1}(\mathbf{f}(1)-\mathbf{f}(2))}.$$

For $i = 1,2$ the coordinates of $\mathbf{f}(i)$ are non-negative integers of which the sum is equal to $\frac{1}{2}n$. Moreover, we assumed that $p > \frac{1}{2}n$. Hence the coordinates of $\mathbf{f}(1)-\mathbf{f}(2)$ have absolute values less than $p$. Together with Lemma 5 this shows that $\mathbf{f}(1)=\mathbf{f}(2)$. But this means exactly that $V_j(P_1)$ and $V_j(P_2)$ are equivalent. □

## 7.5. Proof of $Pr[E_{1j}] \leqslant R \times \binom{n}{\frac{1}{2}n}^{-1}$

Consider an attack $\mathfrak{A}$ on the credential mechanism in which $Z$ does not cheat, and suppose that during some execution of $\mathfrak{A}$ a pseudonym $P$ is properly validated at a moment $t_P$, at which none of the $b$-th roots $(PU_k^{-1})^{b^{-1}}$ $(k = 1, \ldots, R)$ is computable by the individuals. We have to prove that this can happen with probability at most $R \times \binom{n}{\frac{1}{2}n}^{-1}$. In the proof of this fact we need some further notation which is introduced below.

We recall that $t_k$ is the moment at which step 11 of $I(i_k, A_j)$ is executed, and that $A_{k1}, \ldots, A_{kn}$ are the numbers which $i_k$ sent to $Z$ in step 5 of $I(i_k, A_j)$. We define $s_k$ as the moment at which step 8 of $I(i_k, A_j)$ is executed, that is the moment at which $i_k$ shows $R_l$ and $S_l$ to $Z$ for $l \in \mathcal{S}_k$. The stochastic partial function $f_k : \{1, \ldots, n\} \to \mathbb{N}$ is defined as follows: if at moment $s_k$ the individuals can compute $R_l, S_l$ such that $A_{kl} = F(U_k(M_l R_l^a)^b) S_l^{pq}$ then we put $f_k(l) = j$, where $j$ is the positive integer determined by $F_j = F(U_k(M_l R_l^a)^b)$. If the individuals can not compute such $R_l$ and $S_l$ then we do not define $f_k(l)$. $f_k$ is well-defined, that is at moment $s_k$ the individuals can compute at most one pair $(R_l, S_l)$ for each $A_{kl}$. For suppose that at moment $s_k$ the individuals can compute $R_{1l}, S_{1l}$ and $R_{2l}, S_{2l}$ with $A_{kl} = F_1 S_{1l}^{pq} = F_2 S_{2l}^{pq}$, where $F_i = F(U_k(M_l R_{il}^a)^b)$ for $i = 1,2$. Then at moment $s_k$ the individuals can compute $(F_1 F_2^{-1})^{(pq)^{-1}} = S_{1l} S_{2l}^{-1}$. By applying Lemma 5 we obtain $F_1 = F_2$, whence $R_{1l} = R_{2l}$, $S_{1l} = S_{2l}$, as required.

In the first step of the proof we show that $f_k$ is *injective*. Suppose that $f_k$ is defined in both $l$ and $m$, where $l$ and $m$ are distinct integers in $\{1, \ldots, n\}$, and that $f_k(l) = f_k(m)$. Then at moment $s_k$ the individuals can compute $R_l$ and $R_m$ such that $U_k(M_l R_l^a)^b = U_k(M_m R_m^a)^b$. But this implies that $(M_l M_m^{-1})^{a^{-1}} = R_l R_m^{-1}$. Hence the individuals can compute $(M_l M_m^{-1})^{a^{-1}}$ at moment $s_k$. But this is impossible. For since all elements of $\mathfrak{L}$ received by the individuals are of the form $D^{e^{-1}}$, where $D$ was computable by the individuals before this message was received and $e$ is an integer coprime with $a$, all elements of $\mathfrak{L}$ ever computable by the individuals must be of the form

$U^{\mathbf{u}}F^{\mathbf{f}}M^{\mathbf{m}}H^{\mathbf{h}}$, where all coordinates of **u**, **f**, **m** and **h** have denominators coprime with $a$.

In the second step of the proof we show that for $k=1,\ldots,R$, $\mathcal{S}_k$ is statistically independent of $\mathbf{x}_j$, $\mathbf{y}_j$ and $f_j$ for $1\leqslant j\leqslant k$ and $\mathcal{S}_j$ for $1\leqslant j<k$. The steps in a subprotocol are executed at consecutive moments. In particular, $Z$ chooses $\mathcal{S}_k$ at moment $s_k-2$, uniformly from the collection of subsets of $\{1,\ldots,n\}$ of cardinality $\frac{1}{2}n$, and independently of the steps executed at or before moment $s_k-2$. Together with (23), this proves that $\mathcal{S}_k$ is independent of $\mathbf{x}_j$ and $\mathbf{y}_j$ for $1\leqslant j\leqslant k$ and $\mathcal{S}_j$ for $1\leqslant j<k$. By definition, the set of elements of $\mathfrak{X}$ which is computable by the individuals at moment $s_k$ is equal to the set of elements of $\mathfrak{X}$ which is computable from $\mathcal{K}$ and the elements of $\mathfrak{X}$ received by the individuals before moment $s_k$. The elements of $\mathfrak{X}$ sent to the individuals at moment $s_k-1$ are all roots of elements of $\mathfrak{X}$ which were computable by the individuals at moment $s_k-2$. Hence the partial function $f_k$ is completely determined by the set of messages executed at or before moment $s_k-2$. This proves that $\mathcal{S}_k$ is also independent of $f_1,\ldots,f_k$.

For $k=1,\ldots,R$ we put $\mathcal{U}_k=f_k(\{1,\ldots,n\})$, and $\mathcal{W}_k=\mathcal{U}_1\cup\cdots\cup\mathcal{U}_k$. It is easy to check that $\mathcal{T}_k=f_k(\mathcal{S}_k)$, where $\mathcal{T}_k$ is the set defined in (25). From the injectivity of $f_k$ it follows that $\xi_k=1$ (the check in step 9 of $I(i_k,A_j)$ does not fail) if and only if $\#\mathcal{T}_k=\frac{1}{2}n$ and that the vector $\mathbf{e}(\mathcal{T}_k)$, defined in (25), is equal to $(e_1,e_2,\cdots)$, where $e_j=1$ if $j\in\mathcal{T}_k$ and $e_j=0$ otherwise. If $\mathcal{R}$ is some integral domain, $\mathcal{U}$ a finite subset of $\mathbb{N}$, and $\mathbf{y}=(y_1,y_2,\cdots)\in\mathcal{R}^{\infty}$, then we denote by $\mathbf{y}(\mathcal{U})$ the tuple $(\tilde{y}_1,\tilde{y}_2,\cdots)$ with $\tilde{y}_i=y_i$ if $i\in\mathcal{U}$ and $\tilde{y}_i=0$ otherwise.

In the third step of the proof we shall show that there exist an integer $T$ with $1\leqslant T\leqslant R$ and integers $m_k$ for $1\leqslant k\leqslant T$ such that $m_k\xi_k$ is not divisible by $q$ for at least one $k$ in $\{1,\ldots,T\}$ and

$$\sum_{k=1}^{T} m_k\xi_k(\mathbf{y}_k(\mathcal{W}_T)-\mathbf{e}(\mathcal{T}_k))\equiv 0 \bmod q\,. \tag{31}$$

Let $V_j(P)=(V_P,W_P,T_{2P},\ldots,T_{\frac{1}{2}n,P})$ be an $A_j$-validator for $P$, let $\mathcal{T}$ be the set of integers $j$ such that $F_j=F(PT_{lP}^{b})$ for some $l$ in $\{1,\ldots,\frac{1}{2}n\}$, where $T_{1P}=1$. Define the vector $\mathbf{f}(\mathcal{T})\in\tilde{\mathbf{Q}}^{\infty}$ by

$$F^{\mathbf{f}(\mathcal{T})}=\prod_{l=1}^{\frac{1}{2}n}F(PT_{lP}^{b})\,.$$

Since $\gcd(p,q)=1$, there are rational integers $\alpha,\beta$ with $\alpha p+\beta q=1$. It follows (cf (22)) that

$$\tilde{V}:=F^{q^{-1}\mathbf{f}(\mathcal{T})}=(VP_P^2P^{-1})^{\alpha}F^{\beta\mathbf{e}(\mathcal{T})}$$

can be computed by the individuals at moment $t_P$. By Lemma 4, eq. (28), with $h=R$, there are $\tilde{m}_1,\ldots,\tilde{m}_R\in\tilde{\mathbf{Q}}$ such that

$$\sum_{k=1}^{R}\tilde{m}_k\xi_k(\mathbf{y}_k-\mathbf{e}(\mathcal{T}_k))\equiv p\,\mathbf{f}(\mathcal{T}) \bmod q\,. \tag{32}$$

Since $p$ and $q$ are distinct primes larger than $\frac{1}{2}n$ and the coordinates of $\mathbf{f}(\mathcal{T})$ are non-negative and have sum $\frac{1}{2}n$, not all $\tilde{m}_k\xi_k$ are integers divisible by $q$. Let $T$ be the smallest integer with $1\leqslant T\leqslant R$ such that (32) is solvable with $\tilde{m}_k\xi_k\equiv 0 \bmod q$ for $k>T$. Then

$$\sum_{k=1}^{T} \tilde{m}_k \xi_k (\mathbf{y}_k - \mathbf{e}(\mathfrak{T}_k)) \equiv p\mathbf{f}(\mathfrak{T}) \bmod q \ , \tag{33}$$

and $\tilde{m}_T \xi_T$ is not divisible by $q$. By Lemma 4 with $h = T$, $\tilde{V}$ is not computable by the individuals before moment $t_T$, so definitely not at moment $s_T$. But since the individuals can compute $\tilde{V}$ at moment $t_P$, we have

$$s_T < t_P. \tag{34}$$

If the sets $\mathfrak{W}_T$ and $\mathfrak{T}$ would have an element in common, then the individuals were able to compute $R_1$ and $R_2$ at moment $s_T$ such that $U_k R_1^b = P R_2^b$ for some $k \leqslant T$. Together with (34) this would imply that before moment $t_P$ the individuals can compute $(PU_k^{-1})^{b^{-1}}$, which is against our assumption. Hence $\mathfrak{W}_T \cap \mathfrak{T} = \varnothing$. By applying this to (33), using that the coordinates of $\mathbf{f}(\mathfrak{T})$ are 0 on the places outside $\mathfrak{T}$ and considering only the coordinates of both sides of (33) on the places in the set $\mathfrak{W}_T$, and multiplying each $\tilde{m}_k$ with $d$, where $d$ is the smallest positive integer $d$ such that all $d\tilde{m}_k$ are integers for $k = 1, \ldots, T$, we obtain that (31) is satisfied with $m_k := d\tilde{m}_k$ for $1 \leqslant k \leqslant T$, and that $m_k \xi_k$ is not divisible by $q$ for at least one $k$ in $\{1, \ldots, T\}$.

In the fourth step we shall prove that

$$\mathfrak{U}_i \cap \mathfrak{U}_h = \varnothing \text{ for } 1 \leqslant i < h \leqslant R \ . \tag{35}$$

Assume that (35) is false and let $i$ and $h$ be integers with $1 \leqslant i < h \leqslant R$ such that $\mathfrak{U}_i$ and $\mathfrak{U}_h$ have an element in common. Then at moment $s_h$ the individuals can compute $R_1$ and $R_2$ such that $U_i R_1^b = U_h R_2^b$. But this implies that at moment $s_h$, so definitely before moment $t_h$, they can compute $U^{p^{-2}(\mathbf{d}_i - \mathbf{d}_h)}$. Together with Lemma 4 this shows that there are $m_1, \ldots, m_{h-1}$ in $\tilde{\mathbf{Q}}_p$ with

$$\sum_{k=1}^{h-1} m_k \xi_k (\mathbf{d}_k + pq^{-1}\mathbf{x}_k) \equiv \mathbf{d}_i - \mathbf{d}_h \bmod p^2 \ .$$

By comparing the $h$-th coordinates on both sides of this equation and reducing them modulo $p$, we obtain that $0 \equiv -1 \bmod p$, which is impossible. Hence our assumption that (35) is false was wrong.

Our assertion that $Pr[E_{1j}] \leqslant R \times (\frac{n}{2n})^{-1}$ follows at once by combining the results of the previous four steps with Lemma 6 below with $K = GF(q)$.

**Lemma 6.** *Let $K$ be a field and let $\mathbf{y}_1, \ldots, \mathbf{y}_R$, $f_1, \ldots, f_R$, $\mathbb{S}_1, \ldots, \mathbb{S}_R$ be stochastic variables, defined on the same probability space, such that*
*$\mathbf{y}_1, \ldots, \mathbf{y}_R$ assume there values in $K^\infty$;*
*$f_1, \ldots, f_R$ are injective partial functions : $\{1, \ldots, n\} \to \mathbb{N}$ such that the sets $\mathfrak{U}_k := f_k(\{1, \ldots, n\})$ are pairwise disjoint;*
*$\mathbb{S}_1, \ldots, \mathbb{S}_R$ are subsets of $\{1, \ldots, n\}$;*
*for each $k$, the distribution of $\mathbb{S}_k$ is uniform on the collection of subsets of $\{1, \ldots, n\}$ of cardinality $\frac{1}{2}n$ and independent of $\mathbf{y}_k, f_k$ and $\mathbf{y}_j, f_j, \mathbb{S}_j$ for $1 \leqslant j < k$.*
*For $k = 1, \ldots, R$, let $\mathfrak{W}_k = \mathfrak{U}_1 \cup \cdots \cup \mathfrak{U}_k$, $\mathfrak{T}_k = f_k(\mathbb{S}_k)$,*

$\xi_k = 1$ *if* $\#(\mathfrak{T}_k) = \frac{1}{2}n$ *and* $\xi_k = 0$ *otherwise,*
*and* $\mathbf{e}(\mathfrak{T}_k) \in K^\infty$ *the vector of which the coordinates are 1 on the places in* $\mathfrak{T}_k$ *and 0 on the places outside* $\mathfrak{T}_k$.
*For* $T = 1, \ldots, R$, *let* $X_T$ *be the event that there are* $m_1, \ldots, m_T \in K$ *such that* $m_k \xi_k \neq 0$ *for at least one* $k$ *with* $1 \leqslant k \leqslant T$ *and*

$$\sum_{k=1}^{T} m_k \xi_k (\mathbf{y}_k(\mathfrak{W}_T) - \mathbf{e}(\mathfrak{T}_k)) = 0 . \tag{36}$$

*Then*

$$Pr[X_1 \cup \cdots \cup X_R] \leqslant R \times \binom{n}{\frac{1}{2}n}^{-1} .$$

**Proof.** For $T = 1, \ldots, R$ let $X_T^c$ the event that $X_T$ does not take place. It suffices to prove that

$$Pr[X_1] \leqslant \binom{n}{\frac{1}{2}n}^{-1} \text{ and } Pr[X_T \cap X_{T-1}^c] \leqslant \binom{n}{\frac{1}{2}n}^{-1} \text{ for } T = 2, \ldots, R, \tag{37}$$

since

$$Pr[X_1 \cup \cdots \cup X_R] \leqslant Pr[X_1] + \sum_{T=2}^{R} Pr[X_T \cap X_{T-1}^c] .$$

We shall prove (37) only for $T > 1$; the argument for $T = 1$ is even easier and is therefore omitted. Fix $T$ in $\{2, \ldots, R\}$ and denote by $W$ the stochastic tuple consisting of $\mathbf{y}_j$, $f_j (j \leqslant T)$ and $\mathbb{S}_j (j < T)$. Let $\mathfrak{C}$ be the set of values for $W$ for which (36) has a non-trivial solution (i.e. a solution with $m_k \xi_k \neq 0$ for at least one $k$ in $\{1, \ldots, T\}$) but the system

$$\sum_{k=1}^{T-1} m_k \xi_k (\mathbf{y}_1(\mathfrak{W}_{T-1}) - \mathbf{e}(\mathfrak{T}_k)) = 0 \text{ in } m_1, \ldots, m_{T-1} \in K \tag{38}$$

has only solutions with $m_k \xi_k = 0$ for $k = 1, \ldots, T-1$. Fix $w$ in $\mathfrak{C}$ and denote the entries of $w$ by $\mathbf{y}_j, f_j$ $(1 \leqslant j \leqslant T)$ and $\mathbb{S}_j$ $(1 \leqslant j < T)$. Let $m_1, \ldots, m_T$ be a non-trivial solution of (36). By (38), $m_T \xi_T$ is non-zero. Hence $\mathfrak{U}_T$ has cardinality at least $\frac{1}{2}n$. Moreover, there are $m'_1, \ldots, m'_{T-1} \in K$ such that

$$\sum_{k=1}^{T-1} m'_k \xi_k (\mathbf{y}_k(\mathfrak{W}_T) - \mathbf{e}(\mathfrak{T}_k)) = \mathbf{y}_T(\mathfrak{W}_T) - \mathbf{e}(\mathfrak{T}_T) . \tag{39}$$

By combining this with the fact that the sets $\mathfrak{U}_k$ are pairwise disjoint, and considering only the coordinates on the places in $\mathfrak{W}_{T-1}$, we obtain

$$\sum_{k=1}^{T-1} m'_k \xi_k (\mathbf{y}_k(\mathfrak{W}_{T-1}) - \mathbf{e}(\mathfrak{T}_k)) = \mathbf{y}_T(\mathfrak{W}_{T-1}) . \tag{40}$$

If (40) could be satisfied by two different tuples $(m'_k \xi_k : k = 1, \ldots, T-1)$, then a subtraction of these tuples would yield a non-trivial solution of (38) which does not exist by assumption. Hence $m'_1 \xi_1, \ldots, m'_{T-1} \xi_{T-1}$ are uniquely determined by (40). But this implies that the set $\mathfrak{T}_T$ is

uniquely determined by (39). By using that $f_k$ is injective, we obtain that $\mathbb{S}_T$ is uniquely determined by (39). But $\mathbb{S}_T$ is uniformly distributed on the collection of $\binom{n}{\frac{1}{2}n}$ subsets of $\{1, \ldots, n\}$ of cardinality $\frac{1}{2}n$, and independently of $W$. Hence $Pr[X_T \mid W=w] \leq \binom{n}{\frac{1}{2}n}^{-1}$ for each $w \in \mathcal{C}$. Therefore

$$Pr[X_T \cap X_{T-1}^c] = \sum_{w \in \mathcal{C}} Pr[X_T, W=w] = \sum_{w \in \mathcal{C}} Pr[X_T \mid W=w] Pr[W=w] \leq \binom{n}{\frac{1}{2}n}^{-1}.$$

This proves (37). □

## 8. POSSIBLE EXTENSIONS

In this section we briefly mention some extensions of the credential mechanism presented in this paper.

An advantage of this credential mechanism is its flexibility. It has only one major restriction: the set of credentials must be public and fixed before validators are issued, and the amount of computation required also depends linearly on the cardinality of this set. A credential mechanism presented in [Ch 84], which is a variant of that considered here, solves these problems in a natural way. For that mechanism, it is possible to prove an analogue of Theorem 2. A result as strong as Theorem 1 does not hold, but instead one can prove that in essence almost no information about the relationship between pseudonyms used with different organizations is revealed.

We also considered a variation on the credential mechanism that differs mainly in that the validators shown to the organizations are just RSA-signatures on products of the values of the one-way function. For this variation we were able to prove an analogue of Theorem 2 which is not quite as tight as the result in this paper. This variation has the advantage that it can be easily extended—without loss of security for the organizations or privacy of the individuals— to a credential mechanism in which each I-set and O-set can have a restricted number of pseudonyms in common which may be larger than 1. Such an extension may be useful in practice.

In [Ch 84], several ways were presented to build more elaborate and potentially useful structures from these basic credential mechanisms, but the security of such constructions is as yet unproved.

**Acknowledgement**

The authors thank Bert den Boer and Jeroen van de Graaf for their comments and for their suggestions to improve the presentation.

***References***

[Ch 84] D. Chaum, *Showing credentials without identification: transferring signatures between unconditionally unlinkable pseudonyms.* Preprint, available from the author.

[Ch 85] D. Chaum, *Security without identification: transaction systems to make big brother obsolete.* Communications of the ACM , 28(10), Oct. 1985

[DLM 82] R.A. DeMillo, N.A. Lynch, M.J. Merritt, *Cryptographic protocols.* In Proc. 14th ACM Symposium on Theory of computing, pp. 383-400. ACM, 1982.

[EGS 85] S. Even, O. Goldreich, A. Shamir, *On the security of ping-pong protocols when implemented using the RSA.* Presented at Crypto 85, Santa Barbara, August 1985.