Technical University of Denmark

Department of Applied Mathematics and Computer Science

Formal modeling and analysis of privacy of an e-Health protocol

Wenhao Li - s134620

June 2021

Contents

Abstract

It is challenging to formalize and verify privacy properties of cryptographic protocols, and it is desirable to have specifications that are both formally precise and yet simple and intuitive and this may give rise to discover problems that have not yet been discovered earlier due to complexity. The recently proposed method of (α, β)-privacy[1] aims at making models and proofs simpler. This framework has been applied at modeling and proving[2] properties of the voting protocol FOO92. This includes the crucial property of receipt-freeness: a voter cannot prove to a third party, how they have voted. The main motivation is to avoid corruption and coercion of voters.

Also in other privacy-sensititive areas, receipt-freeness is interesting. For instance, in an eHealth system where a doctor can make electronic presciptions for patients, we similarly want a receipt-freeness property to prevent coercion between pharmaceutical companies and doctors. The goal of this thesis is to formally model and analyze an existing e-Health protocol[3] that was made with the receipt-freeness in mind. The modeling will be made using the alpha-beta-privacy approach. This involves conceptual work in formalizing the components and their privacy properties.

1 Introduction

This thesis project is to formalize a proposed privacy-preserving protocol design of an Electronic Health System. We will use a formal method called (α, β)-privacy. The up to date syntax allows modeler to intuitively reasoning about privacy properties of security protocols which has designed with privacy requirements.

The contribution of the thesis are as follows:

1. In the section 2, we will introduce the (α, β)-privacybasis which would be applied in the protocol specification.
2. We introduce the e-Health protocol by first describe how would the e-Health protocol use the cryptographic primitives and how we model them as each component to be used in section 3.
3. A large part of the e-Health protocol is designed with zero-knowledge proof and signed proof of knowledge. And we define a new notation for them with examples so they can be used intuitively in the protocol specification in section 4.
4. Before the protocol specification we briefly go through the analysis of communication channel for the e-Health protocol in section 5.
5. We will specify the protocol and how would the model processed and some counter examples of the privacy goal violation during message exchange, in section 6.

Finally, we will make the conclusion of the thesis.

2 (α, β)-privacy formal basis

The background of (α, β)-privacy is the model applies logical reasoning in First-Order Logic with the Herbrand Universe. The protocol verification are modeled with free algebra and quotient algebra.

2.1 Herbrand Logic

Σ is finite alphabet such as follows:

$$\Sigma = \Sigma_f \uplus \Sigma_i \uplus \Sigma_r$$

where

- Σ_f is a set of uninterpreted function symbol.
- Σ_i is a set of interpreted function symbol.
- Σ_r is a set of relation symbol.

Σ_0 is contains only the uninterpreted constants where $\Sigma_0 \subset \Sigma$. Σ_{op} is the alphabet represents the cryptographic operators, where $\Sigma_{op} \subseteq \Sigma_f$:

2.1.1 Herbrand Logic Terms

$$\begin{aligned} \mathcal{T}_\Sigma(\mathcal{V}) ::= &\ x \\ &\mid f(t_1, \ldots, t_n) \end{aligned}$$

$\mathcal{T}_\Sigma(\mathcal{V})$ represent our Herbrand Logic term where can be either a variable symbol such that $x \in \mathcal{V}$ and $\mathcal{V} \cap \Sigma = \emptyset$; or a functional term such that $f \in \Sigma, \mathsf{arity}(f) = n$, and $n \geq 0$. A term $\mathcal{T}_\Sigma$ is ground if it contains no variable symbol. We use round brackets for uninterpreted functional terms and ommit the brackets when the functional term's arity is zero, i.e. uninterpreted constant such as follows:

$$\mathcal{T}_\Sigma = \begin{cases} c & \textit{if } n = 0 \\ f(t_1, \ldots, t_n) & \textit{if } n > 0 \end{cases}$$

We use square brackets for interpreted functional terms and keep the brackets for the interpreted constants such as follows:

$$\mathcal{T}_\Sigma = \begin{cases} c[] & \textit{if } n = 0 \\ f[t_1, \ldots, t_n] & \textit{if } n > 0 \end{cases}$$

2.1.2 Atom

$$\begin{aligned}\psi ::= &\ p(t_1, \ldots, t_m)\\ &\mid s = t\end{aligned}$$

The atomic formula ϕ is either a predication on the Herbrand Logic terms $\mathcal{T}_\Sigma(\mathcal{V})$ with the $p \in \Sigma_r, \mathsf{arity}(p) = m$ and $m \geq 0$; or an equality of two terms where $s, t \in \mathcal{T}_\Sigma(\mathcal{V})$.

2.1.3 Formula

$$\begin{aligned}\mathcal{L}_\Sigma(\mathcal{V}) ::= &\ \psi && \text{(atomic formula)}\\ &\mid \neg\psi && \text{(negation)}\\ &\mid \mathcal{L}_\Sigma(\mathcal{V}) \wedge \mathcal{L}_\Sigma(\mathcal{V}) && \text{(conjunction)}\\ &\mid \mathcal{L}_\Sigma(\mathcal{V}) \vee \mathcal{L}_\Sigma(\mathcal{V}) && \text{(disjunction)}\\ &\mid \mathcal{L}_\Sigma(\mathcal{V}) \Rightarrow \mathcal{L}_\Sigma(\mathcal{V}) && \text{(implication)}\\ &\mid \mathcal{L}_\Sigma(\mathcal{V}) \Leftrightarrow \mathcal{L}_\Sigma(\mathcal{V}) && \text{(Bi-implication)}\\ &\mid \forall x.\mathcal{L}_\Sigma(\mathcal{V}) && \text{(for all)}\\ &\mid \exists x.\mathcal{L}_\Sigma(\mathcal{V}) && \text{(exists)}\end{aligned}$$

The formula is either an atom, the negation of the atom, logical connectives between two formulas or quantification of formula.

2.2 Herbrand Universe and Algebra.

The uninterpreted function is modeled by the function symbol Σ_f where the functions can represent the desired properties of cyrptographic functions with equational theory which defined by set of identities i.e algebraic properties. It is least congruence[4] relation on the terms $\mathcal{T}_{\Sigma_\mathrm{f}}(\mathcal{V})$ where the terms are closed under substituion and the algebraic properties as follows:

$$[\![t]\!]_\approx = \{t' \in \mathcal{T}_{\Sigma_f} | t' \approx t\}$$

The set of all equivalence classes is our Herbrand Universe U. The quotient term $\mathcal{T}_{\Sigma_\mathrm{f}}$ in the Herbrand Universe is free Σ_f algebra $\mathcal{A}$ consits the equivalence classes as carrier and the function symbols are associated with functions such as following.

$$\begin{aligned}&U = \{[\![t]\!]_\approx | t \in \mathcal{T}_{\Sigma_f}\}\\ &f^\mathcal{A} : U^n \to U, n \geq 0\\ &i.e.\ f^\mathcal{A}([\![t_1]\!]_\approx, \ldots, [\![t_n]\!]_\approx) = [\![f(t_1, \ldots, t_n)]\!]_\approx\end{aligned}$$

2.3 Interpretation

The interpreted function symbols in Σ_i and the relation symbols in Σ_r are associated with functions and relations as well as follows

$$\begin{aligned}\mathcal{I}(f) : U^n \to U\text{for } f \in \Sigma_i \\ \mathcal{I}(r) \subseteq U^n\text{for } r \in \Sigma_r\end{aligned}$$

2.4 Frame

$$\mathsf{F} = \{| m_1 \mapsto t_1, \ldots, m_l \mapsto t_l |\}$$

A frame is a substitution which represents the intruders' memory locations are substituted with protocol messages. The domain of of the frame are the substitutions such that each term is not substituted by itself. Th range of the terms should not contain any of the memory locations because the linear constant restriction.

$$\begin{aligned}\mathcal{D}om(\mathsf{F}) &= \{m | \mathsf{F}(m) \neq m\} \\ \mathcal{R}an(\mathsf{F}) &= \{\mathsf{F}(m) | m \in \mathcal{D}om(\mathsf{F})\} \\ \mathcal{V}ar(\mathcal{R}an(\mathsf{F})) &= \bigcup_{m \in \mathcal{D}om(\mathsf{F})} \mathcal{V}ar(\mathsf{F}) \\ \mathcal{D}om(\mathsf{F}) \bigcap \mathcal{V}ar(\mathcal{R}an(\mathsf{F})) = \emptyset\end{aligned}$$

Recipe is least set contains memory location and closed under Σ_{op} and the syntax is such as follows:

$$\mathit{Recipe}(: \mathcal{T}_\Sigma(\mathcal{V})) \times \mathit{Frame}((\mathcal{V} \to \mathcal{T}_\Sigma(\mathcal{V}))) \to \mathcal{T}_\Sigma(\mathcal{V})$$

2.5 Axioms

The ϕ_{gen} means for every generalization of recipe holds *iff* the recipe is in the domain or exists recipe elements such that they can be formed with function symbol in Σ_{op} and each recipe element is also generalize-able. The ϕ_{hom} means for all recipe elements the generalization of the each recipe element implies the interpretation of conr and the interpretation of struct over the domain. The $\phi_\sim$ for generalize-able two recipes, their concrete knowledge is equal *iff* their structural knowledge is the equal.

$$\begin{aligned}\phi_{gen} \equiv & \forall r. gen(r) \Leftrightarrow \\ & \big(r \in D \vee \bigvee_{f^n \in \Sigma_{op}} \exists r_1, \ldots, r_n. \\ & r = f(r_1, \ldots, r_n) \wedge gen(r_1) \wedge \ldots \wedge gen(r_n)\big) \\ \phi_{hom} \equiv & \bigwedge_{f^n \in \Sigma_{op}} \forall r_1, \ldots, r_n. gen(r_1) \wedge \ldots \wedge gen(r_n) \Rightarrow\end{aligned}$$

$$\begin{aligned}
&\mathsf{concr}[f(r_1,\ldots,r_n)] = f(\mathsf{concr}[r_1],\ldots,\mathsf{concr}[r_n])\wedge\\
&\mathsf{struct}[f(r_1,\ldots,r_n)] = f(\mathsf{struct}[r_1],\ldots,\mathsf{struct}[r_n])\\
\phi_{\sim} \equiv\,&\forall r,s.gen(r)\wedge gen(s)\Rightarrow\\
&\mathsf{concr}[r] = \mathsf{concr}_{\mathrm{s}}[\Leftrightarrow]\mathsf{struct}[r] = \mathsf{struct}_{\mathrm{s}}[]
\end{aligned}$$

concr and struct over memory locations.

$$\begin{aligned}
\phi =\,&\mathsf{concr}[m_1] = t_1\wedge\ldots\wedge\mathsf{concr}[m_n] = t_n\wedge\\
&\mathsf{struct}[m_1] = t_1\wedge\ldots\wedge\mathsf{struct}[m_n] = t_n\wedge
\end{aligned}$$

2.6 α formula

The carrier of α is Σ_0, $\alpha\in\mathcal{L}_{\Sigma_0}\mathcal{V}$. The atom of the term is constructed with the alphabet in Σ_0 and it does not contain any function symbol from Σ_{op}, for instance

$$\alpha\equiv\ x\in\{c_1,\ldots,c_n\}$$

2.7 β formula

The carrier of β is Σ_0, $\beta\in\mathcal{L}_{\Sigma}\mathcal{V}$. The atom of the term is constructed with the alphabet in Σ.

$$\beta\equiv\ \phi_{gen}\wedge\phi_{hom}\wedge\phi_{\sim}$$

2.8 (α,β)-privacy

$$\beta\models\alpha\ \textit{iff}\ \mathcal{I}(\beta)\models\mathcal{I}(\alpha)\text{ for }\mathit{fv}(\alpha)\subseteq\mathit{fv}(\beta)$$

2.9 (α,β)-privacy violation

$$\beta\models\alpha'\wedge\beta\models\alpha\ \textit{iff}\ \alpha\not\models\alpha'\text{ for }\mathit{fv}(\alpha)\subseteq\mathit{fv}(\beta)$$

3 DLVV08 Protocol Description

The proposed e-health protocol aims to protect the privacy of participants like patients and doctors. Therefore we will focus on the commitunications where the two participants are involved. And we will also focus on describing all the details such as what cryptogrphic primitives has been used and how we can represent those cryptographic messages in a formal way with the (α, β)-privacy[1]. Because these are important for protocol specifications in the next chapter.

3.1 Role

The participants of the e-Health protocol are patients, doctors, pharmacist, Medical Prescription Administration, Health Insurance Institute, Social Security Organization, Medical Certification Authority and Central government-approved Certificate Authority. We will use italic font alphabets with upper-case alphabet initials to represent those participants, for instance:

$$A, B, C, M, H, R$$

may represent those participants mentioned above. The two Certification Authorities are stand for trusted government in the e-Health protocol. We will use sans-serif font alphabets with lower-case to represent the trusted authorities. For e.g. s can be used for representing the Certification Authorities. For simplicity, we can assume there is exactly one trusted third party, because they always follow the protocol. We will repeatedly use *role* to represent the finite sorts of the participants of the e-Health protocol.

3.2 ID, Pseudonyms

The *role*s such as doctors and patients, possess some privacy sensitive credential attributes. These credential attributes include *role*s' identifiers and pseudonyms and so on. In e-Health protocol, the link between identifier and pseudonym of doctors are required to be only known to the College of Physicians. We have privacy statements of the form as follows:

$$DrID = \{d_1, \ldots, d_n\}$$

where $DrID$ is the domain of doctor identifiers and $\{d_1, \ldots, d_n\}$ are the constants represents the identifiers. We have privacy statements of the form such as follows:

$$\star A \in DrID$$

This represents the intruder only gets the information of A is one of the doctors in the domain. And the $\star$ means the information would be released in the formula α.

The *role*s such as doctors and patients are assigned with pseudonyms which are generated by seed value of *role*s' identifiers. We normally use italic font alphabets with lower-case initial to represent the function symbol. We will use a function symbol *nym* and a variable A as a seed to map the identifier to a unique pseudonym. For instance,

$$nym(A)$$

represents doctors pseudonyms. For the patients', we should have a disjoint set of constants to represent the domain of patients' identifiers. The patients' pseudonyms can be represent with the same way such as B. In the e-health protocol, the patients also own other credential attributes such as Health Insurance ID and Social Security Status. These attributes can also be represented in a similar way by using other function symbol over variable, for instance:

$$hi(B), ss(B)$$

The terms above represents the Health Insurance ID and Social Security Status respectively.

3.3 Certificate

For the *role*s such as pharmacists, Medical Prescription Administration (MP-A), Health Insurance Institute(HII) and Social Security Organization, the proposed protocol uses X.509 certificate to authenticate these organizations and the certificates are issued by government-approved Certification Organizations. In genneral, a digital certificate can be represented as digital signature. The pharmacist's certificate can be represented as a digital signature of Certification Authority signed on the identifier of the pharmacist for instance:

$$sign(priv(\mathsf{s}), cert(C, mpa(C)))$$

$sign/2, priv/1, cert2, mpa/2$ are function symbols, where $priv(\mathsf{s})$ is the private key of the Certification Authority, C represents the identifier of the pharmacist and $mpa(C)$ represents the pharmacist's MPA identifier. It is an identifier for recognizing which local Medical Prescription Administration does the pharmacist belong. A public key of Certification Authority such $pub(\mathsf{s})$ can be used at retrieving or verifying the signed message. We will use the following algebraic properties to represent that.

$$rsig(sign(priv(s), t)) \approx t$$
$$vsig(pub(s), sign(priv(s), t)) \approx true$$

where $s, t \in \mathcal{T}_{\Sigma_{op}}$. The properties are regarded to the congruence relation induced by equational theories. For instance, with the property in the set of equational theory, for two terms such as $rsig(sign(priv(x_1), cert(x_2, mpa(x_2))))$ and x_2 where $x_1, x_2 \in \mathcal{V}$ and $rsig/1$, $sign/2 \in \Sigma_{op}$, then for a substitution $\sigma = \{x_1 \mapsto \mathsf{s}, x_2 \mapsto \mathsf{a}\}$ we have

$$\sigma(sign(priv(x_1), cert(x_2, mpa(x_2)))) = \sigma(x_2)$$

3.4 Public-key encryption

The e-health protocol uses public-key encryption for the communication with the *role*s such like Medical Prescription Administration(MPA), Health Insurance Institute(HII) and Social Security Organization(SSS). We will use a function symbol to represent the public-key for a *role* for e.g.,

$$pub(M)$$

where $pub/1 \in \Sigma_{op}$ and M is a *role* represents the MPA. Then the public-key encryption can be represented as follows:

$$crypt(pub(p), r, t)$$

where $crypt/3 \in \Sigma_{op}$ and $p, r, t \in \mathcal{T}_{\Sigma_{op}}$. We also need to provide destructor with the algebraic property to represent the decryption for instance,

$$dcrypt(priv(p), crypt(pub(p), r, t)) \approx t$$

where $dcrypt/2 \in \Sigma_{op}$. With the algebraic properties in a set of equational theory, the two terms such as $dcrypt(priv(x_1), crypt(pub(x_1), x_2, x_3))$ and x_3 are unifiable where $x_1, x_2, x_3 \in \mathcal{V}$, with a substitution such as $\sigma = \{x_1 \mapsto \mathsf{m}, x_2 \mapsto \mathsf{r}, x_3 \mapsto \mathsf{t}\}$ then we have:

$$\sigma(dcrypt(priv(x_1), crypt(pub(x_1), x_2, x_3))) = \sigma(x_3)$$

We use the property to represent *role*s can decrypt the public-key encryption with corresponding private-key. The $priv/1 \in \Sigma \setminus \Sigma_{op}$ is a private function for *role*s to generate the private-key with their seed value.

3.5 Commitment

The e-health protocol uses commitment scheme to conceal doctor's or patient's identifier in the commitment phase. The commitment key is chosen randomly by the *role*s in the commitment phase. In the opening phase, the key is used for opening the committed attributes and these attributes should be the same as they were in the prior commitment phase. We can use a function symbol $commit/2 \in \Sigma_{op}$ to represent the commitment scheme for e.g.

$$commit(k, v)$$

where $k, v \in \mathcal{T}_{\Sigma_{op}}$. In the e-health protocol, doctor sends the opening key to the patient to reveal their pseudonym. Therefore we also need to provide a destructor to represent the opening of the commitment. For instance.,

$$open(k, commit(k, v)) \approx v$$

where $open/2 \in \Sigma_{op}$. The algebraic property shows the congruence relation in a set of educational theories. For two terms such as $open(x_1, commit(x_1, x_2))$ and

x_2 where $x_1, x_2 \in \mathcal{V}$ and $open/2, commit/2 \in \Sigma_{op}$, then for a substitution such as $\sigma = \{x_1 \mapsto \mathsf{k}, x_2 \mapsto \mathsf{b}\}$ we have:

$$\sigma(open(x_1, commit(x_1, x_2))) = \sigma(x_2)$$

We use the property to represent *role*s can retrieve a committed value by using correct key. In the e-health protocol, the commitment scheme is used for committing identifiers or pseudonyms of the doctors and patients, i.e. $commit(K, X)$ and $commit(K, nym(X))$ represents those respectively where K, X are variables represent key and *role*.

3.6 Anonymous Credential

The e-health protocol uses anonymous credential for *role*s such as doctors and patients to authenticate themselves to the other *role*s without revealing the identity. The Certification Authority issues anonymous credential based on assertion of *role* credential attributes. This means the assertion should be done anonymously; the credential attributes should be proved; the proof should leak no other information besides the proof. i.e, zero-knowledge proof of credential attributes. We will leave the zero-knowledge for now, because we will explain it with our new notation of modeling the zero-knowledge proof in the next section. To capture the character of the credential attributes, we use the public function to construct the credential with some terms which representing the credential attributes, for example:

$$\begin{aligned}&credD(A, nym(A))\\&credP(B, nym(B), hi(B), ss(B))\end{aligned}$$

where as $credD2, credP/3 \in \Sigma_{op}$ and $A, B \in \mathcal{V}$. The credentials contain different attributes regarding to the different *role*s. For example, doctors have the identifier and pseudonym as their credential attributes, on the another hand patient has some other attributes such as Health Insurance identifier and Social Security identifier.

The reason we chose the function to be public is because, the credential attributes can be selectively disclosed for different proof without submit the entire credential attributes. Therefore, we should provide destructor for the *role*s to retrieve their own credential attributes. The e-Health protocol respect to the fact that Colleague of Physicians knows the correspondence between the doctors' identifier and pseudonyms. For example:

$$\begin{aligned}&getName(credD(A, nym(A))) \approx A\\&getNym(credP(B, nym(B), hi(B), ss(B))) \approx nym(B)\end{aligned}$$

The assertion of the credential attributes by the Certification Authority can be represented as that the Certification Authority's signature on the credential attributes. For example:

$$sign(priv(\mathsf{s}), credD(A, nym(A)))$$

$$\mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B), \mathit{hi}(B), \mathit{ss}(B)))$$

where the two terms represent anonymous credential for doctors and patients.

4 Zero-knowledge Proof

In the e-Health protocol, the *role* such like doctors and patients show their anonymous credential to the recipient for authentication. The protocol also requires a zero-knowledge proof which proves the possession of an anonymous credential by the *role*s without revealing any other information. Now we introduce the new notation for zero-knowledge proofs to make it easy to work with in the protocol description. We describe the syntax of this notation with a context-free grammar and the semantics by translating to an algebraic model, i.e., a set of constructors, destructors and verifiers together with a set of algebraic properties between them. Then we can just use our notation in the (α, β)-privacy process syntax easily without illustrating all the representation regarding to certain Algebraic properties of the checks done by the recipient of zero-knowledge proofs.

$$\begin{aligned}
\mathit{Constant} &::= \mathsf{[a\text{-}z]}(\mathsf{[a\text{-}zA\text{-}Z0\text{-}9]})^* \\
\mathit{Variable} &::= \mathsf{[A\text{-}Z]}(\mathsf{[a\text{-}zA\text{-}Z0\text{-}9]})^* \\
\mathit{Term} &::= \mathit{Constant} \\
&\quad \mid \mathit{Variable} \\
&\quad \mid \mathit{Constant}\ \text{‘(’}\ \mathit{Term}(,\ \ \mathit{Term})^*\text{‘)’} \\
\mathit{PrivateVals} &::= \mathit{Variable}(,\ \ \mathit{Variable})^* \\
\mathit{PublicVal} &::= \mathit{PrivateVal} \\
&\quad \mid \mathit{Variable}\text{“} := \text{”}\,\mathit{Term} \\
&\quad \mid \mathit{Knowledge} \\
\mathit{PublicVals} &::= \mathit{PublicVal}(,\ \ \mathit{PublicVal})^* \\
\mathit{Equation} &::= \mathit{Variable}\ \text{‘=’}\ \mathit{Term} \\
\mathit{Equations} &::= \mathit{Equation} \\
&\quad \mid\ \mathit{Equation}\ \text{‘,’}\ \mathit{Equations} \\
\mathit{Knowledge} &::= \text{“}\mathsf{zkp}\text{”}\ \text{‘\{’}\ \mathit{PublicVals}\ \text{“\}\{”}\ \mathit{PrivateVals}\ \text{‘:’}\ \mathit{Equations}\ \text{‘\}’} \\
&\quad \mid \text{“}\mathsf{spk}\text{”}\ \text{‘\{’}\mathit{PublicVals}\ \text{“\}\{”}\ \mathit{PrivateVals}\ \text{‘:’}\ \mathit{Equations}\ \text{‘\}’}
\end{aligned}$$

The letters in italic are non-terminal symbols. *Constant* are the letters with lower-case initial. Letters with upper-case initial represents the *Variable*. *Term* can be either *Constant* or *Variable* or *Constant* with an open bracket, one or more *Term* and a closed bracket. *PrivateVals* can be either one or more *Variable* for e.g., A, B. *PublicVal* can be either a *Variable* or a notation of *Term* assignment to *Variable* or a *Knowledge*. For instance $CA := \mathit{commit}(K, B))$. *PublicVals* can be either one or more *PublicVals*. *Equation* is the notation of a proof. For e.g. $C = crypt(pub(B), R, nym(A))$. *Equations* can be one or more *Equations*. *Knowledge* can be either a zero-knowledge proof or signed proof of knowledge according to the grammar. For e.g.,

$$\mathsf{zkp}\{Z_0, \ldots, Z_n\}\{X_0, \ldots, X_\mathrm{k}.\phi\}$$

is the notation which forms a zero-knowledge proof. The arity of *Knowledge* term can be counted by the sum of *PublicVals* and *PrivateVals* for a function symbol. We need a substitution such as $\sigma = mgu(\phi)$. The substitution applies to all free variables in the algebraic equations. For example:

$$zkp_\phi(Z_0, \ldots, Z_n, X_0, \ldots, X_\mathrm{k})$$

where $zkp/n + k \in \Sigma_{op}$, $Z_1, \ldots, Z_n, X_1, \ldots, X_k \in \mathcal{V}$. The recipient can retrieve the *PublicVlas* which the sender intentionally released to the recipient. We require $Z_i, X_j \in fv(\phi)$ for $i \in \{0 \ldots n\}, j \in \{0 \ldots k\}$. With the notation, we can provide algebraic property for the destructor and as follows:

$$rzkp^i_\phi(zkp_\phi(Z_0, \ldots, Z_n, X_0, \ldots, X_\mathrm{k})) \approx Z_i \text{ where } Z_i, X_j \in \phi$$

The recipients also need to verify the zero-knowledge proof. And we can provide a verifier with algebraic property as follows:

$$vzkp_\phi(Z_0, \ldots, Z_n, zkp_\phi(Z_0, \ldots, Z_n, X_0, \ldots, X_\mathrm{k})) \approx true$$

where $rzkp_\phi/1, vzkp_\phi/n + 1 \in \Sigma_{op}$. With the most general unifier(*mgu*) as follows:

$$\begin{aligned}\sigma = \{&C \mapsto commit(KA, nym(A)), P \mapsto A, K \mapsto KA,\\ &S \mapsto sign(priv(\mathsf{s}), credD(A, nym(A)))\}\end{aligned}$$

we can have the *PublicVals* and verification as follows:

$$\begin{aligned}&\sigma(rzkp^0_\phi(zkp_\phi(C, P, K, S))) = \sigma(C)\\ &\sigma(vzkp_\phi(C, zkp_\phi(C, P, K, S))) = \sigma(true)\end{aligned}$$

Example-1: (1) Assume Alice has received a zero-knowledge proof of Bob's Diffie-Hellman half key already. (2) Then Alice creates a fresh seed value and construct a zero-knowledge proof of her seed value possession. (3) Then Bob would construct the shared-key and a encrypted message. Alice's process can be illustrated as follows:

(1) ...
(2) νX .
snd (zkp$\{PubA := exp(g, X)\}\{X.Pub_A = exp(g, X)\}$) .
(3) ...

We can observe that the notatin “ := ” in $PubA := exp(g, X)$, it means $PubA$ is referred to the term Pub_A. Our zero-knowledge semantic construct a zero-knowledge function such as follows:

$$zkp_a(exp(g, X), X)$$

Bob would retrieve the public value from the zero-knowledge proof by the following destructor and verifier.

$$rzkp_a^0(zkp_a(exp(g, X), X)) \approx exp(g, X)$$
$$vzkp_a(exp(g, X), zkp_a(exp(g, X), X)) \approx true$$

Bob’s process is illustrated as follows:

```
(1) ...
(2) rcv ( zkp$\{PubA := exp(g, X)\}\{X.Pub\_A = exp(g, X)\}$ ).
(3) let $PubA := rzkp_a^0(zkp_a(exp(g, X), X))$
      if $vzkp_a(exp(g, X), zkp_a(exp(g, X), X))$ == True
      $\nu Msg$ .
      snd( $scrypt(exp(PubA, X), Msg)$ . 0
```

In the Alice’s process, she generate a seed value X, then she sends her zero-knowledge proof to Bob. When Bob receives the zero-knowledge proof, he cannot really see X, but only $PubA$ and may only use it from the received message in the following process. However, the property $Pub\_A = exp(g, X)$ means that Bob does not accept just anything, but performs the verification of the zero-knowledge proof. He need to retrieve the public value from the zero-knowledge proof and use the verifer to perfom the check. After that he can create a new message and use $PubA$ to constructed shared key for symmetric encryption then sends to Alice. The example shows an idea of how we will use the notation in the protocol specification.

4.1 Signed Proof of Knowledge

The e-Health protocol uses signed proof of knowledge for doctor’s to generate the prescription. The prescription includes doctor’s commitment of doctor’s pseudonym, commitment of patient’s pseudonym, prescription text and the prescription identifier. These information is publicly released. Moreover, the commitment of doctor’s pseudonym is proved such that it is the same as the doctor’s pseudonym of its credential attributes. In principle, there is no difference compare to the zero-knowledge proof. In our notation for both of them, the public values are bound to the proof. So for our notation spk the proof can be seen as a signature on the public values by whoever made the proof. In this case this has to be a doctor.

Example-2: In the following process shows a doctor sends her signed-proof of knowledge to patient.

$$\begin{aligned}
&\nu KA\ .\\
&\mathsf{snd}\ (\ \mathsf{spk}\{CA := \mathit{commit}(KA, \mathit{nym}(A))\}\{\\
&\qquad\qquad KA, A, CD.\\
&\qquad\qquad CD = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credD}(A, \mathit{nym}(A))),\\
&\qquad\qquad CA = \mathit{commit}(KA, \mathit{nym}(A))\})\ .\ 0
\end{aligned}$$

Our notion of signed proof of knowledge translate this into an algebraic model where the constructor is as follows:

$$\mathit{spk}_a(\mathit{commit}(KA, \mathit{nym}(A)), KA, A, \mathit{sign}(\mathit{priv}(s), \mathit{credD}(A, \mathit{nym}(A))))$$

where $\mathit{spk}_a \in \Sigma_{op}$ and terms $KA, A \in \mathcal{V}$. patient can retrieve the publicly released information and check the prescription by the following destructor and verifier.

$$\begin{aligned}
&\mathit{rspk}^0_a(\mathit{spk}_a(\mathit{commit}(KA, \mathit{nym}(A)),\\
&\qquad\qquad KA, A, \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credD}(A, \mathit{nym}(A))))) \approx \mathit{commit}(KA, \mathit{nym}(A))\\
&\mathit{vspk}_a(\mathit{commit}(k, \mathit{nym}(t)),\\
&\qquad \mathit{spk}_a(\mathit{commit}(KA, \mathit{nym}(A)),\\
&\qquad\qquad KA, A, \mathit{sign}(\mathit{priv}(s), \mathit{credD}(A, \mathit{nym}(A))))) \approx \mathit{true}
\end{aligned}$$

where $\mathit{rspk}_a/1, \mathit{vspk}_a/2 \in \Sigma_{op}$. In the protocol specification, receving the following term would mean the check of the signed-proof of knowledge with the destructor and verifier which are shown in the above.

$$\begin{aligned}
&\mathsf{rcv}\ (\ \mathsf{spk}\{CA := \mathit{commit}(KA, \mathit{nym}(A))\}\{\\
&\qquad\qquad KA, A, CD.CD = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credD}(A, \mathit{nym}(A))),\\
&\qquad\qquad CA_a = \mathit{commit}(KA, \mathit{nym}(A))\}\ )\ .\ 0
\end{aligned}$$

In the example the doctor first choose a fresh random number KA to build the commitment of pseudonym $\mathit{nym}(A)$. Then the doctor builds the zero-knowledge proof with CA as a public value and KA, A as private value together with his doctor certificate CD. She proves that the public CA is a commitment (to which she knows the secret KA) that commits to the same value $\mathit{nym}(A)$ as in her certificate CD.

4.2 Verifiable Encryption

The e-Health protocol uses verifiable encryption for patient to asymmetrically encrypt patient's pseudonym then sends the encryption to the pharmacist. The pharmacist should not be able to decrypt the message but the encrypted data is proved such that it is exactly the same as the patient's credential attributes without revealing any other information. Furthermore, pharmacist would send the verifiable encryption to other organization for different purpose. We can use our zero-knowledge notation to represent the verifiable encryption.

Example-3: The patient sends a verifiable encryption of his pseudonym to the pharmacist.

$$\begin{array}{l} M \in D_{mpa}\ .\ \nu RA\ . \\ \mathsf{snd}\ (\mathsf{zkp}\{\,\mathit{VN} := \mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B))\}\{ \\ \qquad\qquad B, RA, CD.CD = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B))), \\ \qquad\qquad \mathit{VN} = \mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B))\}\ )\ .\ 0 \end{array}$$

The notation constructs verifiable encryption as follows:

$$\mathit{zkp}_v(\mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)), B, RA, \mathit{sign}(\mathit{priv}(s), \mathit{credP}(B, \mathit{nym}(B))))$$

where $\mathit{zkp}_v/1, \mathit{venc}\ /3, \mathit{ss}/1 \in \Sigma_{op}$ and $M, RA, B, CD \in \mathcal{V}$. The pharmacist can retrieve the verifiable encryption and verify the zero-knowledge proof of the verifiable encryption with following destructor and verifier.

$$\begin{array}{l} \mathit{rzkp}_v^0(\mathit{zkp}_v(\mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)), \\ \qquad\qquad B, RA, \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B)))))) \\ \approx \mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)) \\ \mathit{vzkp}_v(\mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)), \\ \qquad \mathit{zkp}_v(\mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)), \\ \qquad\qquad B, RA, \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B)))))) \\ \approx \mathit{true} \end{array}$$

where $\mathit{rzkp}_v/1, \mathit{vzkp}_v/2 \in \Sigma_{op}$. When pharmacist receives the following term, he would check the zero-knowledge proof of the verifable encryption with the algebraic properties above.

$$\begin{array}{l} \mathsf{rcv}\ (\ \mathsf{zkp}\{\,\mathit{VN} := \mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B))\}\{ \\ \qquad\qquad B, CD.CD = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credD}(B, \mathit{nym}(B))), \\ \qquad\qquad \mathit{VN}_a = \mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B))\}\ )\ . \\ \mathsf{let}\ \mathit{VENC} = \mathit{rzkp}_v^1(\mathit{zkp}_v(\mathit{venc}(\mathit{pub}(\mathsf{s}), RA, \mathit{nym}(B), \\ \quad B, RA, \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B))))\ . \\ \quad \mathsf{if}\ \mathit{vzkp}_v(\mathit{VENC}, \mathit{zkp}_v(\mathit{venc}(\mathit{pub}(M), RA, \mathit{nym}(B)), \\ \qquad B, RA, \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B))))) == \mathsf{True}\ . \\ \quad \mathsf{then}\ \ldots\ .\ 0 \end{array}$$

In the example, the patient choose random value for the verifiable encryption. Then the patient sends a zero-knowledge proof of it to the pharmacist. The zero-knowledge proof proves the encrypted social security status is exactly the same as the one of patient's anonymous credential attributes. After the zero-knowledge proof, the pharmacist will retrieve the publicly released data that the pharmacist cannot decrypt. Then the pharmacist checks the verifiable encryption with the verifier. If the verification passes then the pharmacist continue with the further protocol actions.

5 Protocol Communication Channel

We will describe the communication channel before go to the protocol specification. In general, different channels are used in complex systems, such as authentication, confidential, secure channel and pseudonymous channel. The authentication channel has the property of recognizing the message senders' identity with digital signatures. The confidential channel has the property of only the receiver can decipher the message. The property is achieved by asymmetrically encrypt the message with the receivers public-key. The secure channel which possesses both of the properties. In general, the secure channel is implemented with asymmetrical encryption with the senders' digital signature. The pseudonymous channel requires the senders are authenticated by pseudonyms and also requires the property of confidential channel, for example credit card data is transferred in such a channel so the senders' real name is not exploited and intruder cannot decipher the data.

The e-Health protocol which relies on zero-knowledge protocol for anonymous authentication of doctors and patients. We will observe that in a process where doctor sends the opening information which is the key for the patients to open the commitment in the later opening phase. Without the secure channel implementation, the e-Health protocol trivially violates the secrecy goal. In the protocol specification, we will look into the protocol specification and show example or counter example of whether (α, β)-privacy holds and regarding to the privacy and the receipt-freeness goals.

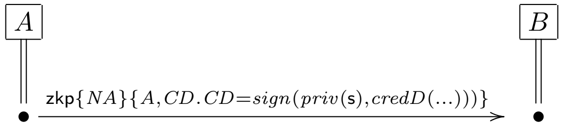
6 Protocol Specification and Analysis

We use the up to date syntax of (α, β)-privacy as much as possible for specifying the e-Health protocol. The protocol initiates by the doctor who starts a new session.

6.1 First process

$$
\begin{aligned}
(1)\ \mathcal{P}_{dr} = & \star A \in \mathit{DrID}.\ \nu\, \mathit{NA}\ . \\
& \mathsf{snd}(\ \mathsf{zkp}\{\mathit{NA}\}\{A, \mathit{CD}.\mathit{CD} = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credD}(A, \mathit{nym}(A)))\}\ )\ .
\end{aligned}
$$

In the first process, The doctor would send a zero-knowledge proof of anonymous credential ownership. First, she has chosen a doctor identifier. $\star A \in \mathit{DrID}$ would release the information to the α i.e., a doctor with an identifier from a domain. She has create a fresh variable for her to use at once in this session and A, CD where CD is her anonymous credential to form the zero-knowledge proof.

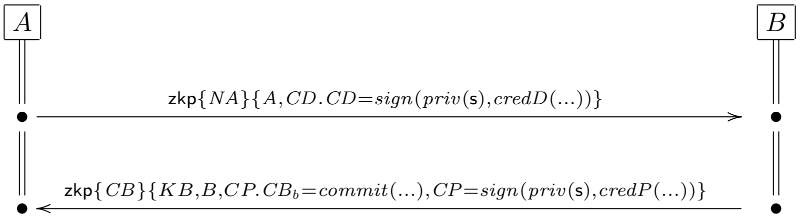


The diagram illustrates the protocol message which contains the high-level of the zero-knowledge proof notation. The patient verifies the zero-knowledge proof with the corresponding constructor,destructor and verifier the same way as the previous examples in the zero-knoledge proof section. We would only see the high-level of the notation here because the notation of zero-knowledge proof works for us respect to the algebraic properties. The patient verifies the zero-knowledge proof and he should only learn the nonce and information released in the α formula.

6.2 Second process

$$\begin{aligned}
(2)\mathcal{P}_{pt} = & \star B \in \mathit{PtID}\ .\nu \mathit{KB}\ . \\
& \mathsf{snd}\ (\ \mathsf{zkp}\{\mathit{CB} := \mathit{commit}(\mathit{KB}, B)\}\{\mathit{KB}, B, \mathit{CP}. \\
& \qquad \mathit{CB}_b = \mathit{commit}(\mathit{KB}, B), \\
& \qquad \mathit{CP} = \mathit{sign}(\mathit{priv}(\mathsf{s}), \mathit{credP}(B, \mathit{nym}(B), \mathit{ss}(B), \mathit{hi}(B)))\}\ )\ .
\end{aligned}$$

In the second process, the patient sends zero-knowledge proof of committed identifier. For this, he has chosen a B as his identifier, again the information would released to the α formula. Then he creates a fresh variable KB as the commitment key. The zero-knowlege proof is formed with the commitment CB as publicly released value and the private value of KB, B, CP whereCP is his anonymous credential.

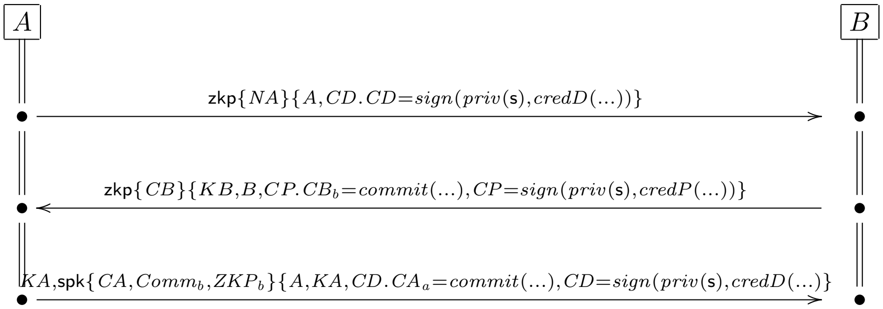

A
B
zkp{NA}{A,CD.CD=sign(priv(s),credD(...))}
zkp{CB}{KB,B,CP.CB_b=commit(...),CP=sign(priv(s),credP(...))}

By the verification of the zero-knowledge proof, the doctor should only learns the patient's commitment and the proof of the commitment.

6.3 Third process

$$
\begin{aligned}
&(3)\ \mathcal{P}_{dr} = vKA\ .\\
&\qquad\qquad \mathsf{snd}(\ KA, \mathsf{spk}\{CA := commit(KA, nym(A)), Comm_b, ZKP_b\}\{\\
&\qquad\qquad\quad A, KA, CD.CA_a = commit(KA, nym(A)),\\
&\qquad\qquad\quad CD = sign(priv(\mathsf{s}), credD(A, nym(A)))\}\ )\ .0
\end{aligned}
$$

In the third process, the doctor sends her commitment key and her signed-proof of knowledge i.e. the prescription. She generates a fresh variable to compute the commitment of her pseudonym. The public value also contains the patient's commitment $Comm_b$ and his zero-knowledge proof ZKP_b. The private value contains the doctors identifier A, commitment key KA and her credential CD which the proof proves released commitment is generalized from her private value.


A
B
zkp{NA}{A,CD.CD=sign(priv(s),credD(...))}
zkp{CB}{KB,B,CP.CB_b=commit(...),CP=sign(priv(s),credP(...))}
KA,spk{CA,Comm_b,ZKP_b}{A,KA,CD.CA_a=commit(...),CD=sign(priv(s),credD(...)}

From the process the patient obtains the commitment key and the prescription. And He also learns a new message such as the doctor's commitment and the proof of the doctor's commitment.

Example-4: In the process, the intruder cannot violate the privacy goal even he obtains the commitment key. For doctors send the prescriptions to patients in two session, the α formula is as follows:

$$\alpha = A_1 \in \mathit{DrID} \wedge A_2 \in \mathit{DrID}$$

where α models two doctors identifiers are picked from the domain. The frame F has been updated with new message as follows:

$$\mathsf{F} = \{|\ \ldots, m_3 \mapsto pair(KA, spk_a(\ldots)), m_{10} \mapsto pair(KA, spk_a(\ldots))|\}$$

where m_3 and m_{10} stores the prescriptions of two sessions. Intruder can form recipes r_1 and r_2 with the public cryptographic functions over his memory locations just like how patient would open the commitment of the doctors' pseudonyms as follows:

$$r_1 = open(proj_1(m_3), rspk_a^0(proj_2(m_3)))$$

$$r_2 = open(proj_1(m_{10}), rspk_a^0(proj_2(m_{10})))$$

Intruder can observe two cases by comparing the concrete knowledge of the two pseudonyms whether they are equal or not. With the β formula for a substitution $\sigma = \{A_1 \mapsto d_1, A_2 \mapsto d_1\}$ we would have

$$\begin{gathered}\mathsf{concr}_3[r_1] = nym(d_1) = \mathsf{concr}_{10}[r_2] \Rightarrow \\ \mathsf{struct}_3[r_1] = nym(A_1) = \mathsf{struct}_{10}[r_2] = nym(A_2)\end{gathered}$$

Forr a substitution $\sigma = \{A_1 \mapsto d_1, A_2 \mapsto d_2\}$ and recipe r_1 and r_2 we would have

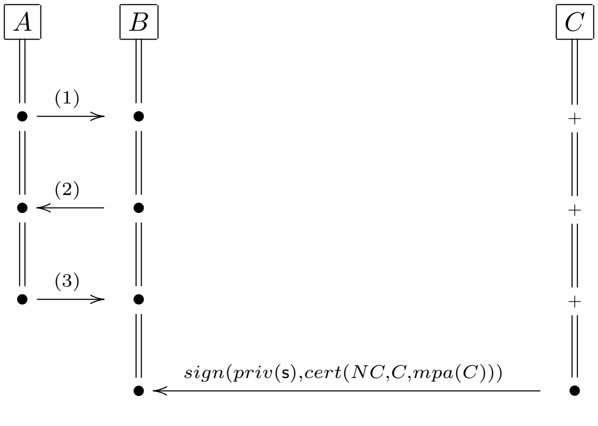
$$\begin{gathered}\mathsf{concr}_3[r_1] = nym(d_1) \neq \mathsf{concr}_{10}[r_2] = nym(d_2) \Rightarrow \\ \mathsf{struct}_3[r_1] = nym(A_1) \neq \mathsf{struct}_{10}[r_2] = nym(A_2)\end{gathered}$$

In the process, we can observe the doctors' pseudonyms are distinguishable because the doctor sends out the opening information without secure channel. The intruder can only find out whether two pseudonyms are the same or different. This does not mean intruder violates the privacy as still $\beta \models \alpha$.

6.4 Fourth process

$$\begin{aligned}(4)\ \mathcal{P}_{ph} =& \star C \in PhID \\ & \mathsf{snd}\ (\ sign(priv(\mathsf{s}), cert(C, mpa(C)))\ )\ .\end{aligned}$$

In the fourth process, the pharmacist picks an identifier from its domain. The information would released in the formula α. Then he should use his certificate to authenticate himself to the patients for whom wishes to send the prescriptions for medicine.



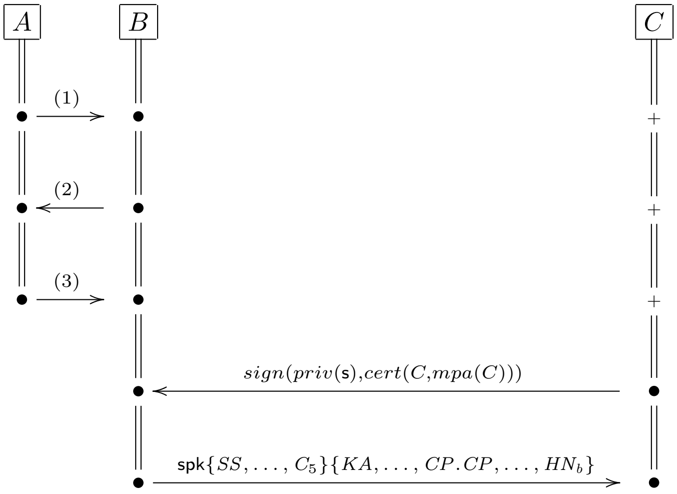
The patient would verify the digital signature and extract the identifier of the pharmacist to generate acknowledgement in the later process.

6.5 Fifth Process

$$
\begin{aligned}
&(5)\mathcal{P}_{pt} = M \in D_{mpa}, R \in D_{ri}, H \in D_{hi}\ .\ \nu\overline{Rv}\ .\\
&\quad \textsf{snd}\ (\textsf{spk}\{SS := ss(B),\\
&\qquad\qquad VC_1 := venc(pub(M), Rv_1, hi(B)),\\
&\qquad\qquad VC_2 := venc(pub(M), Rv_2, nym(A)), Presc_a,\\
&\qquad\qquad VC_3 := venc(pub(R), Rv_3, hi(B)),\\
&\qquad\qquad VC_3^{'} := venc(pub(R), Rv_4, nym(B)),\\
&\qquad\qquad VC_4 := venc(pub(M), Rv_5, nym(B)),\\
&\qquad\qquad C_5 := crypt(pub(M), Rv_7, venc(pub(H), Rv_6, nym(B))\}\{\\
&\qquad\qquad KA, KB, B, Rv_1, Rv_2, Rv_3, Rv_4, Rv_5, Rv_6, CP.\\
&\qquad\qquad CP = sign(priv(\mathsf{s}, credP(B, nym(B), hi(B), ss(B)))),\\
&\qquad\qquad SS_b = ss(B),\\
&\qquad\qquad MH_b = venc(pub(M), Rv_1, hi(B)),\\
&\qquad\qquad MD_b = venc(pub(M), Rv_2, nym(A)),\\
&\qquad\qquad RH_b = venc(pub(R), Rv_3, hi(B)),\\
&\qquad\qquad RN_b = venc(pub(R), Rv_4, nym(B)),\\
&\qquad\qquad MN_b = venc(pub(M), Rv_5, nym(B)),\\
&\qquad\qquad HN_b = venc(pub(H), Rv_6, nym(B))\}\ .
\end{aligned}
$$

In the fifth process, the patient anonymously authenticates to the pharmacist with the signed proof of knowledge. The patient should first pick the identifier of the *role*'s such as M, R and H. Then he chooses fresh variables $\overline{Rv}$ for encryptions. After that he forms the signed knowledge proof the zero-knowledge notation. The knowledge proves his social security status $SS(B)$, the plain text Health Insurance identifier and pseudonym $hi(B)$ and $nym(B)$ in VC_1, VC_3 and VC_4 is the same as his anonymous credential attributes and those plain text will be encrypted with the different *role*s' public-key. The knowledge also proves the doctors' pseudonym $nym(A)$ in verifiable encryption VC_2 is the same as the one in the prescription $Presc_a$ where the prescription is received in the second process.

All verifiable encryptions can only be decrypted by the corresponding recipients with correct private key and also be verified by the correct public-key. But the well-formed zero-knowledge proof of verifiable encryption itself can be verified by any *role*. A verifiable encryption $venc(pub(H), Rv_6, nym(B))$ is meant for the Health Insurance Institute. Because of the e-Health protocol requires the pharmacist should not be able to verify which health institute does the patient belong, instead the verification task is handed over to the MPA. Therefore, the verifable encryption is encrypted C_5 with $pub(M)$ which released in the publicly, moreover it is proved by the signed knowledge proof.


A
B
C
(1)
(2)
(3)
$sign(priv(\mathsf{s}),cert(C,mpa(C)))$
$\mathsf{spk}\{SS, \ldots, C_5\}\{KA, \ldots, CP.CP, \ldots, HN_b\}$

Example-5: If a certain prescribed medicine exceeds the dosage, then pharmacist should ask the patient for releasing the doctor's identifier. Regarding to the issue of intrusive monitoring of doctors' prescription pattern, the doctors' privacy goal depends on the communication channel. The α is defined as follows:

$$\alpha' = A_1 \in DrID \land A_2 = d_1$$

where $A_2 = d_1$ means the doctor's identifier has been released due to the limitation of the medicine dosage. The following frame F' is updated with new messages due to two sessions of the fifth process.

$$\mathsf{F}' = \{| \ldots, m_5 \mapsto spk_b(\ldots), \mapsto d_1, m_{12} \mapsto spk_b(\ldots), m_{13} \mapsto d_1 |\}$$

where m_5 and m_{12} stores patients' signed proof of knowledge. With the frame intruders can form following recipes to extract the public values of the signed proof of knowledge.

$$\begin{aligned} r_3 &= rspk_a^0(rspk_b^3(m_5)) \\ r_4 &= rspk_a^0(rspk_b^3(m_{12})) \end{aligned}$$

where r_3 and r_4 represent the the recipes from two sessions. With the β formula, for a substitution $\sigma = \{A_1 \mapsto d_1\}$ we have following formulae.

$$\begin{aligned} \mathsf{concr}_5[r_3] &= commit(ka, nym(d_1)) \land \\ \mathsf{concr}_{12}[r_4] &= commit(ka, nym(d_1)) \land \\ \mathsf{struct}_5[r_3] &= commit(KA, nym(A_1)) \land \\ \mathsf{struct}_{12}[r_4] &= commit(KA, nym(A_2)) \end{aligned}$$

where (α, β)-privacy holds if the e-Health protocol is implemented with secure channel. But if that is not the case, the doctors' privacy goal is violated.

$$\begin{aligned} r_5 =& open(proj_1(m_3), rspk_a^0(rspk_b^3(m_5))) \\ r_6 =& open(proj_1(m_{10}), rspk_a^0(rspk_b^3(m_{12}))) \end{aligned}$$

With the released doctors' identifier, the intruder can form a pseudonym of the doctor's with the public function $nym/1$ by his own such as:

$$r_7 = nym(m_{13})$$

Then the intruder can derive following formula:

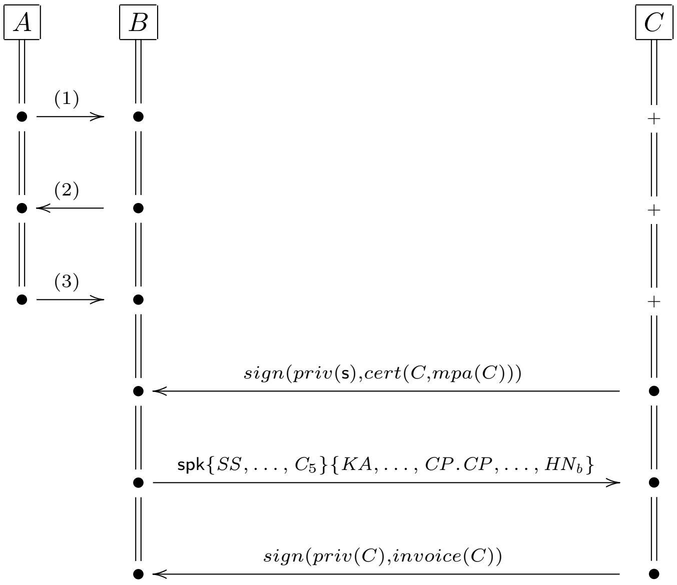
$$\begin{aligned} \mathsf{concr}_5[r_5] = \mathsf{concr}_{12}[r_6] = \mathsf{concr}_{12}[r_7] = nym(d_1) \Rightarrow \\ \mathsf{struct}_5[r_5] = \mathsf{struct}_{12}[r_6] = \mathsf{struct}_{12}[r_7] \Rightarrow \\ A_1 = A_2 \end{aligned}$$

The (α, β)-privacy does not hold in the state because $\beta \models A_1 = A_2$ but $\alpha \not\models A_1 = A_3$. The intruder can derive the formula from the β where the α does not entail the formula, i.e. the intruder can link doctors' pseudonym from prescription by obtaining the released doctor's identifier due to the dosage restriction and the commitment key from the second process without secure channel implementation.

6.6 Sixth Process

$$(6)\ \mathcal{P}_{ph} = \mathsf{snd}(\ sign(priv(C), invoice(C))\ )\ .$$

In the sixth process, the pharmacist issues bill $invocie(C)$ by signing with his private key after he delivers the prescribed medicine to the patient.

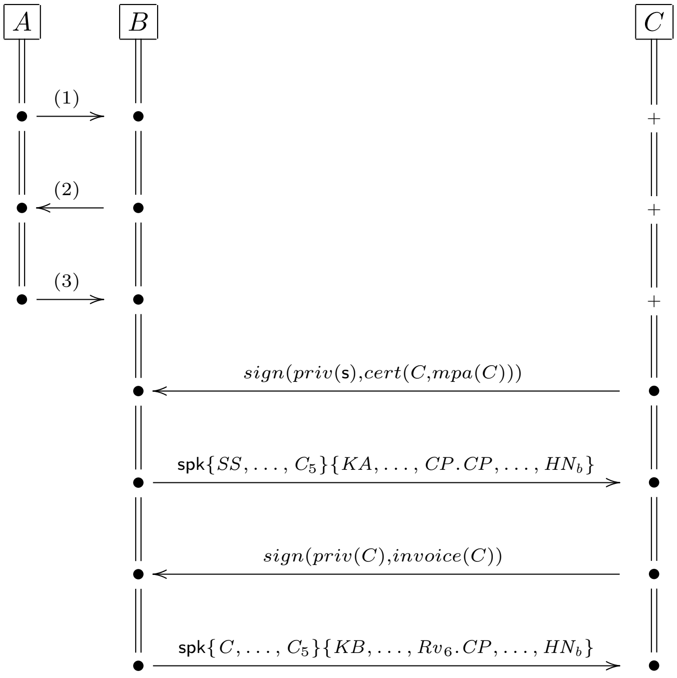

A
B
C
(1)
(2)
(3)
sign(priv(s),cert(C,mpa(C)))
spk{SS, ..., C5}{KA, ..., CP.CP, ..., HNb}
sign(priv(C),invoice(C))

The patient should verify the bill with the pharmacist's public-key.

6.7 Seventh Process

$$\begin{aligned}(7)\ \mathcal{P}_{pt} = \mathsf{snd}\ (\ \mathsf{spk}\{&C, sign(priv(C), invoice(C)), VC_1, VC_2, Presc_a, VC_3, VC_3^{'},\\ &VC_4, C_5\}\{KB, KA, B, Rv_1, Rv_2, Rv_3, Rv_4, Rv_5, Rv_6, CP.\\ &CP = sign(priv(\mathsf{s}, credP(B, nym(B), hi(B), ss(B))))\\ &MH_b = venc(pub(M), Rv_1, hi(B))\\ &MD_b = venc(pub(M), Rv_2, nym(A)),\\ &RH_b = venc(pub(R), Rv_3, hi(B)),\\ &RN_b = venc(pub(R), Rv_4, nym(B)),\\ &MN_b = venc(pub(M), Rv_5, nym(B)),\\ &HN_b = venc(pub(H), Rv_6, nym(B))\}\ )\ .\ 0\end{aligned}$$

In the seventh process, the patient would send signed-proof of knowledge of the invoice to the pharmacist which represents the acknowledgement of the transaction. The public values are composed with the identifier of the pharmacist C and the bill received in the sixth process. Moreover, it is included with the same verifiable encryptions which has been earlier. The private values and proofs are the same as in the fifth process.


A
B
C
(1)
(2)
(3)
$sign(priv(\mathsf{s}),cert(C,mpa(C)))$
$\mathsf{spk}\{SS,\ldots,C_5\}\{KA,\ldots,CP.CP,\ldots,HN_b\}$
$sign(priv(C),invoice(C))$
$\mathsf{spk}\{C,\ldots,C_5\}\{KB,\ldots,Rv_6.CP,\ldots,HN_b\}$

With the patient's acknowledgement, the pharmacist can continue further actions with the Medical Prescription Administration.

7 Conclusion

We have formalized the components by means of the e-Health protocol cryptographic primitives with Algebraic properties in the protocol description. DLVV-08 has beend designed with zero-knowledge proof, so we have defined a new notion for that. We have showed how to model and verify the e-Health protocol with (α, β)-privacy process syntax by defining the privacy sensitive identifiers in the α formula and use the β formula to analyze the technical information.

1. The e-Health protocol has proposed doctors' identity should not be link-able by any other participants except the College of Physicians. By our analysis, the protocol violates the doctors' privacy in the fifth process. The intruder would retrieve the doctor's pseudonym with the commitment opening information. Furthermore, he can analyze whether the doctors' identifier are link-able to the pseudonym if there is a medicine dosage issue arises.

2. The e-Health protocol has proposed that the e-Health should protect doctors privacy regrading to the doctors, they should not be able to issue provable evidence of a doctor's prescription behaviour by not associating the doctors' real identity in the prescription text, i.e. receipt-freeness goal. We have model the recipt-freeness does not hold in the fifth process. The intruder can have the doctor's identifier if there is a dosage issue arises. Then the pharmacists have a ability compare the identifier with pseudonyms from the prescription, i.e. signed proof of knowledge. Because of the signed proof of knowledge is also verifiable by intruder, the property is substantially a proof of a doctor's pseudonym.

3. The e-Health protocol has proposed the e-Health protocol should resist the coercion of the doctors' prescription. Although, we have not modeled the coercion-resistance in the report, but we can briefly discuss about how to model that. This involves the model of the prescription pattern, i.e. we need define the medicine name as privacy sensitive variable in the α and analyze them in the β formula.

References

[1] S. Mödersheim and L. Viganò, “Alpha-beta privacy,” *ACM Trans. Priv. Secur.*, vol. 22, no. 1, pp. 7:1–7:35, 2019.

[2] S. Gondron and S. Mödersheim, “Formalizing and proving privacy properties of voting protocols using alpha-beta privacy,” in *Computer Security - ESORICS 2019 - 24th European Symposium on Research in Computer Security, Luxembourg, September 23-27, 2019, Proceedings, Part I* (K. Sako, S. A. Schneider, and P. Y. A. Ryan, eds.), vol. 11735 of *Lecture Notes in Computer Science*, pp. 535–555, Springer, 2019.

[3] B. D. Decker, M. Layouni, H. Vangheluwe, and K. Verslype, “A privacy-preserving ehealth protocol compliant with the belgian healthcare system,” in *Public Key Infrastructure, 5th European PKI Workshop: Theory and Practice, EuroPKI 2008, Trondheim, Norway, June 16-17, 2008, Proceedings* (S. F. Mjølsnes, S. Mauw, and S. K. Katsikas, eds.), vol. 5057 of *Lecture Notes in Computer Science*, pp. 118–133, Springer, 2008.

[4] F. Baader, W. Snyder, P. Narendran, M. Schmidt-Schauss, and K. Schulz, “Chapter 8 - unification theory,” in *Handbook of Automated Reasoning* (A. Robinson and A. Voronkov, eds.), Handbook of Automated Reasoning, pp. 445–533, Amsterdam: North-Holland, 2001.