2003－ＣＳＥＣ－22 (5)
2003／7／17

演繹体系による暗号方式の形式化と体系の性質としての暗号方式の安全性

Ashraf Moustafa Bhery[†] 萩原 茂樹[†] 米崎 直樹[†]

[†] 東京工業大学大学院情報理工学研究科計算工学専攻 〒152–8552 東京都目黒区大岡山 2–12–1

E-mail: †{ashraf,hagihara,yonezaki}@fmx.cs.titech.ac.jp

あらまし 非対称暗号方式等、暗号方式の特徴やその安全性は、確率理論や計算量理論を用いてこれまで盛んに研究されてきた。本稿では、理想化された非対称暗号方式を形式化可能なJDE体系と呼ぶ演繹体系を提案する。この演繹体系を用いることにより、様々な非対称暗号方式の類似した安全性を統一的に取り扱うことが可能となる。本稿では、安全性を表現するためにいくつか新しいアイデアを導入した。まず、*content-of* 等いくつか関数を導入した。これにより、「攻撃者は、二つの暗号文の中身を見ることなく、その中身が等しいかどうか知ることができる」等の状況を形式化するのに十分な推論規則を構成することが可能となった。さらに、判定文に加えて、演繹体系の性質として非判定文を導入した。これらを用いることにより、非対称暗号方式の安全性である「内容区別不能性」「鍵区別不能性」「内容非展性」「鍵非展性」を定義することが可能となった。これらの安全性はそれらの十分条件を示すことにより証明される。「内容長非展性」も同様にこの体系で証明される。JDE体系は、理想状況での攻撃者の可能な振る舞いをすべて捉えており、その意味では、この体系は暗号方式の直観的な完全な公理系である。

キーワード 非対称暗号、判定文、演繹体系、内容区別不能性、鍵区別不能性、内容非展性、鍵非展性、内容長非展性

A new deduction system for cryptographic primitives and their security properties

Ashraf Moustafa BHERY[†], Shigeki HAGIHARA[†], and Naoki YONEZAKI[†]

[†] Tokyo Institute of Technology, Graduate School of Information Science and Engineering,
Department of Computer Science.
Tokyo Meguro Oookayama 2–12–1, 152–8552 Japan
E-mail: †{ashraf,hagihara,yonezaki}@fmx.cs.titech.ac.jp

Abstract The characterization and security properties of cryptographic primitives such as asymmetric encryption schemes have been well developed using the notions of probability and complexity theory. In this paper, we propose a new deduction system called the *JDE-system* which can be used to formalize an idealized asymmetric encryption scheme. In our system, deductive reasoning is used to identify similar security properties of different asymmetric encryption schemes. New functions are introduced for describing several security properties. For example, by using the function *'content-of'*, we can provide a sufficient set of inference rules that are used to formalize facts such as "without seeing the content of ciphertexts, an attacker has the opportunity to see whether two different ciphertexts have the same content". We use the notion of *"Judgement"* in our JDE-system. Conversely, we also introduce the notion of *"Unjudgment"* as a property of JDE-system. By using these notions, we can define the *content-indistinguishability*, *key-indistinguishabilty*, *content-non-malleability*, and *key-non-malleability* of asymmetric encryption schemes. A proof is given showing the sufficient conditions for these security properties. We also clarify the relationships that exist between these security properties. Two new security properties that we call *key-non-malleability*, and *content-length-non-malleability* are proven using the JDE-system. The JDE-system identifies all of the procedures that an attacker could employ. In this sense, the JDE-system is a completely intuitionistic axiomatic realization of an encryption scheme.

Key words asymmetric encryption, judgment, unjudgment, deduction systems, content-indistinguishability, key-indistinguishabilty, content-non-malleability, key-non-malleability, content-length-non-malleability

1. Introduction

In this paper, we propose a new deduction system called the Judgment Deduction System (or the *JDE-system*). Using the JDE-system, we can analyze the security properties of an asymmetric encryption scheme. To accomplish this, we introduce new functions such as *content-of.* Then we define a set of inference rules related to this function. With these rules, we analyze the fact that "without seeing the content of ciphertexts, an attacker has the chance to see whether two different ciphertexts have the same content". A different fact such as "without seeing the content of ciphertexts, an attacker has the opportunity to see whether two different ciphertexts are encrypted with the same key or not" can also be analyzed by using the function *key-of.* We also use the notion of *judgment* in our system. Conversely, we also introduce the notion of *unjudgment* as a property of JDE- system. With these two notions, a proof is given showing the sufficient conditions for the *content-indistinguishability*, *key-indistinguishabilty*, *content-non-malleability*, and *key-non-malleability* of asymmetric encryption schemes. A proof is given to formalize their relationship. New stecurity properties for asymmetric encryption schemes are also proposed. A formal proof of these new properties is provided. For example, *key-non-malleability* is a new security property for an asymmetric encryption scheme. It has not been previously defined for either asymmetric or symmetric encryption schemes. *Content-length-non-malleability* for an asymmetric encryption scheme is a formalization of the fact that an attacker, given an example ciphertext with an unknown content length, finds that it is impossible for him to output a different ciphertext such that their content lengths are related. The *content-length-non-malleability* security property is considered an extension of the popular cryptographic security notion *content-non-malleability* for an asymmetric encryption scheme.

Related work. Many formal approaches have been proposed for the analysis of security protocols including the BAN logic of authentication by Burrows, Abadi, and Needham [11], the CSP with a model checking FDR approach [4]~[6], equation rewriting tools [17], and the inductive approach for the analysis of security protocols by Paulson [3] which provides automated support using his Isabelle proof assistant. Another approach for modeling and reasoning about security protocols is the spi-calculus, introduced by Abadi and Gordon in [9], [10]. Most of these approaches focused on proving the authentication and secrecy properties of security protocols, however, in these formal methods there is a common set of rules which defines the analysis of an attacker's knowledge. For examples, in Paulson approach [3], attacker's knowledge is defined as a set of terms S and the attacker's analysis of the his knowledge S are defined by the fundamental operations $parts(S)$, $analz(S)$, and $synth(S)$. With these analysis some useful properties are obtained such as $synth(analz(S))$ represents the set of terms which the attacker can be obtained and generated from the set of knowledge S. A similar set of inference rules are found in the Schneider approach [4]. In [16], Bolignano describes the basic attacker's operations (encrypt, decrypt, pairing, decomposition of a pair) as a transformation relation from a set of terms S to a set S'. Multiple application of these basic operations to a set of terms S provides a new terms (set or single) S'. Such analysis is used to define the relations S' *known-in* S and S' *comp-of* S. Then some impossibility properties like $\neg(b$ *comp-of* $S) \Rightarrow \neg(b$ *known-in* $S)$ is proved. As we have seen, there is a common set of rules for the analysis of an attacker's knowledge. This common set of rules is also used in our system (see Section 2.1). Our analysis of an attacker's knowledge is not limited to the analysis defined by $T \models T_1$. It also includes an analysis defined by the judgment relation $T \models T_1 \odot T_2$ and the unjudgment relation $T \models^{?} T_1 \odot T_2$, where $\odot \in \{=, \mp\}$.

This paper is organized as follows: Section 2 presents a common asymmetric encryption scheme . We consider this to be a basic semantic of encryption. It is used to construct the JDE-system. New functions that are used for defining the JDE-system and a set of sufficient JDE-inference rules are introduced in Section 3. In Section 4, we state various facts about the JDE-system. Using these facts, various syntactic security notions of an asymmetric encryption scheme are verified in Section 5. Finally we give our conclusions and suggestions for future work in Section 6.

2. A basic asymmetric encryption scheme

In this section, we explain a basic asymmetric encryption scheme based on the current literature for the formal analysis of security protocols [3]~[7], [15]~[17]. We start by defining the primitive data types (sorts) that are used in this paper. **Public-Data**, **Keys**, and **Secrets** are disjoint sets of atomic terms. **Public-Data** is the set of symbols 0 and 1. **Keys** is a nonempty set denoting the set of long-term keys. It consists of the set of public keys **Public-Key** and the set of corresponding private keys **Private-Key**. There is a one-to-one relationship between public keys and their corresponding private keys. We will denote the public keys by $k_1, k_2, \ldots$ and their corresponding private keys by $k_1^{-1}, k_2^{-1}, \ldots$ respectively. If we use only one key, then we will remove the subscript. **Secrets** is a nonempty set of symbols that may have a secret design in accordance with some policy. We will denote the elements of this set by $n_1, n_2, \ldots$. Compound terms can be constructed by either asymmetric encryption or concatenation. We will write **Terms** as a set of terms that is defined as follows:

[Definition 1] A term T is a well formed expression defined as

$$\langle Primitive \rangle \begin{cases} i & \text{for each } i \text{ in } \textbf{Public-Data}; \\ n & \text{for each } n \text{ in } \textbf{Secrets}; \\ k & \text{for each } k \text{ in } \textbf{Public-Key}; \\ k^{-1} & \text{for each } k^{-1} \text{ in } \textbf{Private-Key} \end{cases}$$

$$\langle Compound \rangle \begin{cases} \{T\}_k & \text{for encryption} \\ \{T\}_{k^{-1}} & \text{for digital signature} \\ T_1, T_2 & \text{for concatenation} \end{cases}$$

Informally, $\{T\}_k$ represents the asymmetric encryption of term T using a public key k and $\{T\}_{k^{-1}}$ represents the digital signature of term T using a private key k^{-1}. T_1, T_2 represents the concatenation of the two terms T_1 and T_2. For concatenation, we use parentheses when it is deemed necessary to ensure that only one interpretation can be inferred. We consider that each term T is associated with a natural number $len(T)$ that represents its length. For the encryption scheme, we assume the following conditions:

(1) Encryption is *perfect*. i.e, it satisfies the following:

- It is impossible to produce $\{T\}_k$ without knowing T and k.
- In order to recover T from $\{T\}_k$, one must know k^{-1}.
- Given $\{T\}_k$, it is impossible to recover k.

(2) Signature is *perfect*. This means that the signature satisfies the same conditions as in 1.

(3) Only by seeing T, the attacker can determine the length of T.

(4) If T_1 and T_2 are syntactically different, then the values of T_1 and T_2 are different.

(5) An attacker always distinguishes encrypted terms from non-encrypted terms.

(6) An attacker always distinguishes signature terms from terms of different constructions.

(7) An attacker always distinguishes the concatenation of terms from terms of different constructions.

In the above sense, we consider that our JDE-system is ideal for asymmetric encryption scheme. We do not assume any actual specific cryptosystem, such as RSA, in which the commutative property for encryption is satisfied (this means that in RSA, we have $\{\{T_1\}_{k_1}\}_{k_2} = \{\{T_1\}_{k_2}\}_{k_1}$).

2.1 An attacker's composition and decomposition relationship

In this section, we give the basic attacker's composition and decomposition relationship for an asymmetric encryption. The basic definition was given by Dolev and Yao [14] and has been used in the formal analysis of security protocols [3]~[7], [15]~[17].

[Definition 2] (Basic-rules) Let T_1 and T_2 be two terms. The relation $\models$ over T_1, T_2, denoted by $T_1 \models T_2$, is defined as the least relation that satisfies the following rules:

(1) $\overline{\models i}$ (for each $i \in$ **Public-Data**),

(2) $\overline{\models k}$ (for each $k \in$ **Public-Key**),

(3) $\dfrac{\models T_1}{T \models T_1}$,

(4) $\overline{T \models T}$,

(5) $\dfrac{T \models T_1, T_2}{T \models T_1}, \dfrac{T \models T_1, T_2}{T \models T_2}$,

(6) $\dfrac{T \models T_1, T \models T_2}{T \models T_1, T_2}$,

(7) $\dfrac{T \models T_1, T \models k}{T \models \{T_1\}_k}$,

(8) $\dfrac{T \models T_1, T \models k^{-1}}{T \models \{T_1\}_{k^{-1}}}$,

(9) $\dfrac{T \models \{T_1\}_k, T \models k^{-1}}{T \models T_1}$,

(10) $\dfrac{T \models \{T_1\}_{k^{-1}}, T \models k}{T \models T_1}$.

Intuitively $T_1 \models T_2$ infers not only what an attacker can obtain from T_1 but also which terms an attacker can generate from T_1 with the knowledge of public keys and public data. For example, an attacker can generate terms such as $\models \{0\}_k$,$\models \{0,1\}_k$, and $\models \{1, k_1\}_k$ from the above rules.

3. The JDE-system

In this section, we propose a new inference system for an asymmetric encryption scheme called the Judgment Deduction System (**JDE system**) for an encryption scheme. The JDE system can be used for reasoning about the security properties of cryptographic primitives. It consists of a triple $(\sum, \mathcal{I}, \vdash_{JDE})$, where:

(1) $\sum$ is the set of judgment,

(2) $\mathcal{I}$ is the set of inference rules. Each JDE-inference rule has zero or more premises and a conclusion. A conclusion is said to be *derivable* if all the premises of JDE-inference rule holds.

(3) $\vdash_{JDE}$ is the JDE-derivation relation, which is defined using a sequence of applications of JDE-inference. It is also defined inductively.

3.1 Judgment

In order to define judgment, we must first extend the set **Terms** to the set **Ex-Terms** by introducing the following functions. Then we can define *propositions* over the new set **Ex-Terms**. In the following, *key-for* is a well-known function but the others are new.

- *key-of* : **Terms** $\longmapsto$ **Terms**.

Semantically, it takes an encrypted (digital signature) term and returns the encrypted key (digital signature key).

- *key-for* : **Terms** $\longmapsto$ **Terms**.

Semantically, it takes an encrypted (digital signature) term and returns the key that is needed for decryption.

- *content-of* : **Terms** $\longmapsto$ **Terms**.

Semantically, it takes an encrypted (digital signature) term and returns its content term.

The function *key-for* in [3] is used for analyzing security protocols and their properties. In this paper, we use this function for reasoning about the security properties of encryption schemes. For example, by using the same function we can analyze the fact that attacker has the opportunity to see whether two different digital signature terms request the same key for verification. For an asymmetric encryption scheme, we need a function *key-of*. With this function, we are able to analyze whether the attacker can determine that two ciphertexts are encrypted with the same key without knowing the decryption keys. The new function, *content-of*, is used to analyze whether the attacker has the opportunity to see that two different cipheryexts have the same content without seeing the contents of the actual ciphertexts.

[Definition 3] Let f be a meta variable of the sort $\{$*content-of*, *key-of*, *orkey-for*$\}$. The set of extended terms denoted by ***Ex-Terms*** is defined as follows:

(1) If $T \in$ **Terms**, then $T \in$ **Ex-Terms**,

(2) If $\{T_1\}_k \in$ **Terms**, then $f(\{T_1\}_k) \in$ **Ex-Terms**,

(3) If $\{T_1\}_{k^{-1}} \in$ **Terms**, then $f(\{T_1\}_{k^{-1}}) \in$ **Ex-Terms**,

Next, we define *propositions* over **Ex-Terms** as follows:

[Definition 4] Let T_i and T_j be meta variables that range over the set **Ex-Terms**. $T_i = T_j$ and $len(T_i) = len(T_j)$ are propositions. If P is a proposition, then $\neg P$ is also a proposition.

The proposition $\neg(T_i = T_j)$ is abbreviated as $T_i \neq T_j$. Similarly, the proposition $\neg(len(T_i) = len(T_j))$ is abbreviated as $len(T_i) \neq len(T_j)$.

[Definition 5] (Judgment) Judgment is expressed as $T \models P$, which means that by knowing T, an attacker can see the fact P that is represented by a proposition.

[Definition 6] A statement is defined as an expression of the form $T \models T_1$ or the form $T \models P$. We use meta variables $S_1, S_2, \ldots$ to range over statements.

3.2 JDE-inference rules

The JDE-inference rules are defined by the following set of inference rules. In the following JDE-inference rules, we use $T_1, T_2, \ldots$ to denote meta-variables of sort **Terms**, $T_1', T_2', \ldots$ to denote meta-variables of sort **Ex-Terms**, and $T_1'', T_2'', \ldots$ to denote meta-variables of sort **Ex-Terms** $\cup \{len(T) | T \in$ **Ex-Terms**$\}$.

(1) Basic-rules in Definition 2,

(2) $\dfrac{T \models T_1}{T \models len(T_1)}$,

If an attacker obtains term T_1 from T, then he can determine its length.

(3) $\dfrac{T \models T_1}{T \models T_1 = T_1}$,

If an attacker obtains term T_1 from term T, then he can determine that the values of these terms are equal.

(4) $\dfrac{T \models T_1, T \models T_2}{T \models T_1 \neq T_2}$ (where T_1 is not syntactically equal to T_2),

With the knowledge of term T, if an attacker obtains the two terms T_1 and T_2 with the condition that T_1 and T_2 are syntactically different, then he can determine that these terms have different values.

(5) $\dfrac{T \models \{T_1\}_{k_i} = \{T_2\}_{k_j}}{T \models f(\{T_1\}_{k_i}) = f(\{T_2\}_{k_j})}$,

With the knowledge of T, if an attacker can deduce that the values of two encrypted terms are equal then he can determine that these two encrypted terms have the same content and the same encrypted keys, and that they need the same key for decryption.

(6) $\dfrac{T \models \{T_1\}_{k_i^{-1}} = \{T_2\}_{k_j^{-1}}}{T \models f(\{T_1\}_{k_i^{-1}}) = f(\{T_2\}_{k_j^{-1}})}$,

This is similar to the rule above.

(7) $\dfrac{T \models T_1'' = T_2''}{T \models T_2'' = T_1'',}$,

(8) $\dfrac{T \models T_1'' \neq T_2''}{T \models T_2'' \neq T_1''}$,

These two rules represent the symmetry of judgment.

(9) $\dfrac{T \models T_1'' = T_2'', T \models T_2'' = T_3''}{T \models T_1'' = T_3''}$,

(10) $\dfrac{T \models T_1'' = T_2'', T \models T_2'' \neq T_3''}{T \models T_1'' \neq T_3''}$,

(11) $\dfrac{T \models T_1}{T \models \textit{content-of}(\{T_1\}_k) = T_1}$,

This rule must be true, because even if an attacker does not know the private key k^{-1}, if he knows T_1 and if the public key k is available, he can generate $\{T_1\}_k$. Then, he can determine that the values of the two encrypted terms are equal. Otherwise, if he knows the private key, then this rule is trivially true.

(12) $\dfrac{T \models \{T_1\}_{k^{-1}}}{T \models \textit{content-of}(\{T_1\}_{k^{-1}}) = T_1}$,

This rule is true because an attacker knows the public key k.

(13) $\dfrac{T \models T_1}{T \models \textit{key-of}(\{T_1\}_k) = k}$,

This is similar to rule 11.

(14) $\dfrac{T \models \{T_1\}_{k^{-1}}}{T \models \textit{key-for}(\{T_1\}_{k^{-1}}) = k}$

(where k is the inverse key of k^{-1}),

This rule is true because an attacker knows all of the public keys and there is a one-to-one relationship between the public and private keys.

(15) $\dfrac{T \models \textit{key-for}(T_1) = \textit{key-for}(T_2)}{T \models \textit{key-of}(T_1) = \textit{key-of}(T_2)}$,

This rule is true because there is a one-to-one relationship between public and private keys.

(16) $\dfrac{T \models T_1' = T_2'}{T \models len(T_1') = len(T_2')}$,

If an attacker knows about the equality of the values of two terms, then he can determine the equality of their length.

(17) With the conditions T_1 and T_2 are syntactically different , we have:

(a) $(T_1, T_2, left) \dfrac{T \models T_1, T \models T_2}{T \models len(T_1) = len(T_2)}$,

(b) $(T_1, T_2, right) \dfrac{T \models T_1, T \models T_2}{T \models len(T_1) \neq len(T_2)}$.

From the knowledge of T, if an attacker can obtain T_1 and T_2 from T and they are syntactically different, then he can determine whether they have the same length.

3.3 JDE-derivation

It is clear from the JDE-inference rules above that rule 22 dictates that a choice must be made to apply either rule $(T_1, T_2, left)$ or rule $(T_1, T_2, right)$ in the deduction system. Therefore, various deduction systems can be created using JDE-inference rule 22. For this case, we introduce the following inductive definition of *JDE-derivation*(or $\vdash_{JDE}$). With this definition, we conclude that if one rule from JDE-inference rule 22 is applied for two given terms T_1 and T_2, then we must continue to apply the same rule for the same two terms in any other deductions.

[Definition 7] ($\vdash_R$.) Let us assume that S_1, S_2, and S_3 are meta variables that range over *statement* and that R and R' are meta variables that range over subsets of the set $\mathcal{I} = \{\textit{rule } i \mid 1 \leqq i \leqq 22\}$. Then $\vdash_R$ is defined inductively as follows:

- Base case. If the JDE-inference rule r_1 is of the form $\frac{}{S_1}$, then $\vdash_{\{r_1\}} S_1$.
- Induction. If we have $\vdash_R S_1, \vdash_{R'} S_2$, and there is a JDE-inference rule $r \in \mathcal{I}$ of the form $\frac{S_1, S_2}{S_3}$ (or the form $\frac{S_1}{S_3}$), then we have $\vdash_{R \cup R' \cup \{r\}} S_3$(or $\vdash_{R \cup \{r\}} S_3$).

[Definition 8] The relation $\vdash_{JDE} S_1$ is a JDE-derivation defined as: $(\vdash_R S_1 \wedge R$ satisfies$\neg \exists T_i \exists T_j ((T_i, T_j, left) \in R \wedge (T_i, T_j, right) \in R))$.

For our JDE-system, we propose a set of JDE-inference rules that are sufficient for defining all of the possible actions that can be taken by an attacker. An attacker can decompose, compose, encrypt, and decrypt. In some cases, an attacker can determine the contents of encrypted terms without knowing the secret key and determine the encrypted key from examining the ciphertext. These rules are sufficient for formalizing all of an attacker's abilities. For instance, consider the following rule:

$$\frac{T \models \{T_1\}_{k_i} \neq \{T_2\}_{k_j}, T \models \textit{content-of}(\{T_1\}_{k_i}) = \textit{content-of}(\{T_2\}_{k_j})}{T \models \textit{key-of}(\{T_1\}_{k_i}) \neq \textit{key-of}(\{T_2\}_{k_j})}$$

This rule states that if two terms are syntactically different and have the same content, then these two terms must use different keys for encryption. However, we do not have to add this rule to the JDE-inference rules because the judgment

$$T \models \textit{key-of}(\{T_1\}_{k_i}) \neq \textit{key-of}(\{T_2\}_{k_j})$$

can be deduced in the JDE-scheme if an attacker can deduce the judgments $T \models \{T_1\}_{k_i} \neq \{T_2\}_{k_j}$ and $T \models \textit{content-of}(\{T_1\}_{k_i}) = \textit{content-of}(\{T_2\}_{k_j})$.

For the same reason, we do not have to add the following inference rule.

$$\frac{T \models \{T_1\}_{k_i} \neq \{T_2\}_{k_j}, T \models \textit{key-of}(\{T_1\}_{k_i}) = \textit{key-of}(\{T_2\}_{k_j})}{T \models \textit{content-of}(\{T_1\}_{k_i}) \neq \textit{content-of}(\{T_2\}_{k_j})}.$$

As another example to support our premise that we have a sufficient set of JDE-inference rules, it is not necessary to add the following rule:

$$\frac{T \models k^{-1}, T \models \{T_1\}_{k^{-1}}}{T \models \textit{key-of}(\{T_1\}_{k^{-1}}) = k^{-1}}$$

to deduce the judgment $T \models \textit{key-of}(\{T_1\}_{k^{-1}}) = \textit{key-of}(\{T_2\}_{k^{-1}})$.

This conclusion can be proven by using the following JDE-derivation tree:

$$\frac{\dfrac{T \models \{T_1\}_{k^{-1}}}{T \models \textit{key-for}(\{T_1\}_{k^{-1}}) = k} \quad , \quad \dfrac{\dfrac{T \models \{T_2\}_{k^{-1}}}{T \models \textit{key-for}(\{T_2\}_{k^{-1}}) = k}}{T \models k = \textit{key-for}(\{T_2\}_{k^{-1}})}}{\dfrac{T \models \textit{key-for}(\{T_1\}_{k^{-1}}) = \textit{key-for}(\{T_2\}_{k^{-1}})}{T \models \textit{key-of}(\{T_1\}_{k^{-1}}) = \textit{key-of}(\{T_2\}_{k^{-1}})}}$$

[Definition 9] $\nvdash_R T \models P \overset{def}{\Leftrightarrow} T \models P \notin \{S \mid \vdash_R S\}$.

[Definition 10] Unjudgment: $\dashv_{JDE} T \models^? P \overset{def}{\Leftrightarrow} \forall R(\nvdash_R T \models P \wedge \nvdash_R T \models \neg P)$.

Definition 9 gives a formal description of the statements that an attacker cannot derive. By using this definition, we introduce the definition of an Unjudgment relation, which means that from the term T, the attacker has no evidence in any JDE-derivation to allow him to determine whether the statement $T \models P$ or the statement $T \models \neg P$ is true. The definition of an Unjudgment relation can be considered as the heart of our formalization for the security functions of an asymmetric encryption scheme.

4. Facts about the JDE-system for cryptographic primitives

In this section, we describe various important facts about the JDE-system. These facts can be used in several ways to express several security notions of an asymmetric encryption scheme as shown in the next section. This should be considered as a specific feature of our work.

[Theorem 1] Even if an attacker knows the content of one of the two encrypted terms, he cannot obtain any information about the relationship between the contents of the two encrypted terms.

If $\{T_1\}_{k_i} \neq \{T_2\}_{k_j}$ and ($\nvdash_{JDE} T \models T_1$ or $\nvdash_{JDE} T \models T_2$), then

$$\dashv_{JDE} T \models^? \textit{content-of}(\{T_1\}_{k_i}) = \textit{content-of}(\{T_2\}_{k_j}).$$

[Theorem 2] If $(\{T_1\}_{k_i} = \{T_2\}_{k_j} \wedge \vdash_{JDE} T \models \{T_1\}_{k_i}) \vee (\vdash_{JDE} T \models T_1 \wedge \vdash_{JDE} T \models T_2)$, Then either:

(1) $\vdash_{JDE} T \models \textit{content-of}(\{T_1\}_{k_i}) = \textit{content-of}(\{T_2\}_{k_j})$

or:

(2) $\vdash_{JDE} T \models \textit{content-of}(\{T_1\}_{k_i}) \neq \textit{content-of}(\{T_2\}_{k_j})$ is true.

This means that if the attacker can see the contents of two ciphertexts, then he can know whether these two ciphertexts have the same contents or not. Also if the attacker knows one ciphertext and makes a copy of it (this means that they are syntactically equal), then he can know these two ciphertexts have the same contents.

[Theorem 3] Even if an attacker knows the content of one of the two encrypted terms, an attacker cannot obtain the relationship between the encrypted keys of the two encrypted terms.

If $\{T_1\}_{k_i} \neq \{T_2\}_{k_j}$ and ($\nvdash_{JDE} T \models T_1 \vee \nvdash_{JDE} T \models T_2$) then $\dashv_{JDE} T \models^? \textit{key-of}(\{T_1\}_{k_i}) = \textit{key-of}(\{T_2\}_{k_j})$.

[Theorem 4] If $(\{T_1\}_{k_i} = \{T_2\}_{k_j} \wedge \vdash_{JDE} T \models \{T_1\}_{k_i}) \vee (\vdash_{JDE} T \models T_1 \wedge \vdash_{JDE} T \models T_2)$, then we have either:

(1) $\vdash_{JDE} T \models$ *key-of*$(\{T_1\}_{k_i}) =$ *key-of*$(\{T_2\}_{k_j})$ or:

(2) $\vdash_{JDE} T \models$ *key-of*$(\{T_1\}_{k_i}) \ddagger$ *key-of*$(\{T_2\}_{k_j})$ holds.

The meaning of this theorem is similar to theorem 2.

[Theorem 5] If $\dashv_{JDE} T \models^?$ *content-of*$(\{T_1\}_{k_i}) =$ *content-of*$(\{T_2\}_{k_j})$, then $\dashv_{JDE} T \models^?$ *len*(*content-of*$(\{T_1\}_{k_i})) =$ *len*(*content-of*$(\{T_2\}_{k_j}))$.

[Theorem 6] If $\dashv_{JDE} T \models^?$ *key-of*$(\{T_1\}_{k_i}) =$ *key-of*$(\{T_2\}_{k_j}$, then $\dashv_{JDE} T \models^?$ *len*(*key-of*$(\{T_1\}_{k_i})) =$ *len*(*key-of*$(\{T_2\}_{k_j}))$.

The above two theorems state that if the attacker can not obtain the relationship between the contents (or between the encryption keys) of two ciphertexts, then he cannot obtain any information about the relationship between the length of contents (or length of encryption keys) of the two encrypted terms.

5. Verification of famous security notions using our JDE-system

Various notions exist about the security of an asymmetric encryption scheme. There are also various mechanized proofs, most of which are based on the notions of probability and complexity theory. For example, the most basic is the notion of *probabilistic indistinguishablity* introduced by Goldwasser and Micali [2]. Another notion is called *probabilistic non-malleability* introduced by Dolev, Dwork and Naor [13]. The relation between these notions is defined in [8], [12]. In this section, we give a different formalism for the security notion of an asymmetric encryption scheme without using the notion of probability and complexity theory. In addition, we introduce a new security notion for an asymmetric encryption scheme. Our analysis formally states the conditions that are sufficient for defining these security notions. The first security notion, called *content-indistinguishability* is the most natural one for defining the secrecy property in the *formal analysis of security protocols* [9], [10]. The security notion of *key-indistinguishability* is stated in the work of Abadi and Rogaway [1] for a symmetric encryption scheme in a scenario that differs from that considered in our analysis. The syntactic security notion of *content-non-malleability* has the same meaning as the security notion *non-malleability* in [13]. We have extended the security notion *non-malleability* to the syntactic security notion of *key-non-malleability* . We consider this to be an important feature of our work. From the facts stated in the previous section, the security notions of *content-non-malleability* and *content-indistinguishability* encryption schemes are characterized in corollaries 1 and 3. They are derived from theorem 1. The security notions of *key-non-malleability* and *key-indistinguishability* asymmetric encryption schemes are characterized in corollaries 2 and 4. They are derived from theorem 3. Corollary 5 is derived from theorems 1 and 5. New security properties about length, such as *content-length-non-malleability*, can be defined and characterized by using corollary 1 and theorem 5.

[Corollary 1] (*Content-Non-malleability.*)
If $(\vdash_{JDE} T \models \{T_1\}_{k_i}, \vdash_{JDE} T \models T_2, \nvdash_{JDE} T \models T_1)$, then
$\dashv_{JDE} T \models^?$ *content-of*$(\{T_1\}_{k_i}) =$ *content-of*$(\{T_2\}_{k_j})$.

[Corollary 2] (*Key-Non-malleability.*)
If $(\vdash_{JDE} T \models \{T_1\}_{k_i}, \vdash_{JDE} T \models T_2, \nvdash_{JDE} T \models T_1)$ then
$\dashv_{JDE} T \models^?$ *key-of*$(\{T_1\}_{k_i}) =$ *key-of*$(\{T_2\}_{k_j})$.

The above two state that if an attacker, given an encrypted term whose contents he does not know, finds that it is impossible for him to generate a different encrypted term such that their contents (encrypted keys) are related.

[Corollary 3] (*Content-indistinguishability)*
If $(\vdash_{JDE} T \models \{T_1\}_{k_i}, \vdash_{JDE} T \models \{T_2\}_{k_j}, \{T_1\}_{k_i} \ddagger \{T_2\}_{k_j}, \nvdash_{JDE} T \models T_1, \nvdash_{JDE} T \models T_2)$ then
$\dashv_{JDE} T \models^?$ *content-of*$(\{T_1\}_{k_i}) =$ *content-of*$(\{T_2\}_{k_j})$.

[Corollary 4] (*Key-indistinguishability)*
If $(\vdash_{JDE} T \models \{T_1\}_{k_i}, \vdash_{JDE} T \models \{T_2\}_{k_j}, \{T_1\}_{k_i} \ddagger \{T_2\}_{k_j}, \nvdash_{JDE} T \models T_1, \nvdash_{JDE} T \models T_2)$ then
$\dashv_{JDE} T \models^?$ *key-of*$(\{T_1\}_{k_i}) =$ *key-of*$(\{T_2\}_{k_j})$.

The above two corollaries means that an attacker, given two different encrypted terms whose contents he does not know, finds that it is impossible to distinguish between their contents (or between the encryption keys). By theorem 5, we propose a new security property for asymmetric encryption schemes, called *content-length-non-malleability.*

[Corollary 5] (Content-length-non-malleability)
If $(\vdash_{JDE} T \models \{T_1\}_{k_i}, \vdash_{JDE} T \models$ *len*$(T_2), \nvdash_{JDE} T \models$ *len*(T_1), $\{T_1\}_{k_i} \ddagger \{T_2\}_{k_j}$, then
$\dashv_{JDE} T \models^?$ *len*(*content-of*$(\{T_1\}_{k_i})) =$ *len*(*content-of*$(\{T_2\}_{k_j}))$.

6. Conclusion and future work

In this paper, we have proposed a formal deduction system called the JDE-system. The JDE-system formalizes all of an attacker's abilities, which are stated in the assumptions of an encryption scheme. With this system, we can analyze a case such as "without seeing the content of ciphertexts, an attacker has the opportunity to see whether two different ciphertexts have the same content" by using our new function *content-of.* In the JDE-system, we have introduced the notions of *judgment* and *unjudgment* as meta properties of the JDE-system. With these notions, we have determined the sufficient conditions for asymmetric *content-indistinguishability*, *key-indistinguishabilty*, *content-non-malleability*, and *key-non-malleability* encryption schemes and have provided proofs for these conditions. Our work is closely related to the ongoing efforts towards reducing the gap between formal analysis and modern cryptographic analysis for cryptographic primitives. Extensions to our JDE-system to include a cryptographic primitive hash function are now in progress. Our future work will also include a verification of our work in modern cryptography.

文　献

[1] M. Abadi and P. Rogaway. Reconciling two views of cryptography:The computational soundness of formal encryption. In *IFIP International Conference on Theoretical Computer Science*, August, 2000.

[2] S. Goldwasser and S. Micali. Probabilistic encryption.*Journal of*

Computer and System Science, 28:270-299,April 1984.

[3] L. C.Paulson. The inductive approach to verifying cryptographic protocols.*Journal of Computer Security*,6(1-2):85-128,1998.

[4] S. Schneider. Security Properties and CSP. IN *IEEE Symposium on Security and Privacy*,pages 174-187,1996.

[5] G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In *Proceedings of 10th IEEE Computer Security Foundations Workshop*, pages 45-58,1997.

[6] P. Ryan and S. Schneider. Modelling and Analysis of Security Protocols. *Addison Wesley*,2001.

[7] P. Lincoln, J. Mitchell, M. Mitchell, and A. Scedrov. A probabilistic poly-time framework for protocol analysis. In *Proceedings of the 5th ACM Conference on Computer and Communications Security*, pages 112-121,1998.

[8] M. Bellare and P. Rogaway. Relations among notions of security for public-key encryption schemes. In *Advances in Cryptology - Lecture Notes in Computer Science*. No. 1462, Springer-Verlag, 1998, pages 26-45.

[9] M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The Spi calculus. In *Proceedings of the 4th ACM Conference on Computer and Communications Security*, pages 36-47, 1997.

[10] M. Abadi and A. D. Gordon. A bisimulation method for cryptographic protocols: The Spi calculus. In *Nordic Journal of Computing*, 5(4):267-303, Winter 1998.

[11] M. Burrows, M. Abadi and R. Needham. A logic of authentication. *Proceedings of the Royal Society of London A*, 426:233-271, 1989.

[12] S. Micali, C. Rackoff and B. Sloan. The notion of security for probabilistic cryptosystems. In*SIAM Journal of Computing*, April 1988.

[13] D. Dolev, C. Dwork and M. Naor. Non-malleable cryptography. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, ACM 1991.

[14] D. Dolev, C. Yao. On the security of public key protocols. In *IEEE Transactions and Information Theory*, IT-29(12):189-208,1983.

[15] T. Y. C. Woo and S. S. Lam. A semantic model for authentication protocols. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*, Oakland, CA, May 1993.

[16] D. Bolignano. An approach to the formal verification of cryptographic protocols. In *Proceedingd of the 3th ACM Conference on Communucations and Computer Security (CCS-96)*, pages 106-118, 1996.

[17] R. Kemmerer, C. Meadows, and J. Millen. Three systems for cryptographic protocol analysis. *Journal of Cryptography*, 7(2), 1994.