Relationship between Two Approaches for Defining the Standard Model PA-ness

Isamu Teranishi[†] and Wakaha Ogata[‡]

[†] NEC Corporation.
1753, Shimonumabe, Nakahara-Ku, Kawasaki, Kanagawa, 211-0011, Japan.
[‡] Tokyo Institute of Technology.
2-12-1 Ookayama, Meguro-Ku Tokyo, 152-8550, Japan.
`teranisi@ah.jp.nec.com, wakaha@mot.titech.ac.jp`

Abstract. There are two approaches to define Plaintext Awareness (PA). The first one is a classical approach to define the PA security and is used to define the PA security of the random oracle model. This approach enables us to define the PA-ness simply, but no one know whether we can define the standard model PA security based on this approach. In contrast, the second approach is a current approach to define the PA security. It enables us to define the standard model PA security formally, but it is more elaborate than the overwhelming-based approach. In this paper, we aim to clarify relations between the two approaches. We define the standard model PA security based on the first approach. Then we show that, under a very weak condition, it is equivalent to the known definition of the standard model PA security based on the second approach.

Keywords: Plaintext Awareness, Standard Model.

1 Introduction

1.1 Background

The *Plaintext Awareness (PA)* [BR94,BDPR98,HLM03,BP04,D06,TO06,BD07] is one of the most fundamental notion about a Public-Key Encryption scheme (PKE). Intuitively, we say that a PKE is PA secure, if it satisfies the following property: whenever an adversary generates a ciphertext, the adversary "knows" the corresponding plaintext.

The PA notion is important, because the PA-ness together with the IND-CPA security implies the IND-CCA2 security [BR94,BDPR98,BP04]. This means that we can use the PA security when we show the IND-CCA2 security. Moreover, it can bring some insight or an alternative perspection on the design of existing PKE with IND-CCA2 security, as said by Bellare and Palacio [BP04].

Although the intuitive definition mentioned above is quite simple, it is elaborate task to define the PA notion formally. Therefore, many definitions of the PA security are there. Mainly, there are two approaches to defining PA security, which we will call "overwhelming-based approach" and "indistinguishability-based approach."

The *overwhelming-based approach* is a classical approach to define the PA security and is used to define the PA security [BR94,BDPR98] of the random oracle model. This approach enables us to define the PA-ness simply, but no one know whether we can define the standard model PA security based on this approach. In contrast, the *indistinguishability-based approach* is a current approach to define the PA security. It enables us to define the standard model PA security formally [BP04], but it is more elaborate than the overwhelming-based approach.

Reviewing Two Approaches: Both the overwhelming-based approach and the indistinguishability-based approach are defined by using an adversary and an extractor. However, the details of two approaches are quite different. In the case of the overwhelming-based approach, the adversary outputs one ciphertext and the extractor extracts the corresponding plaintext from the ciphertext. We say that a PKE is PA secure, if there exists an extractor which succeeds the extraction with overwhelming probability.

In contrast, the indistinguishability-based approach defines the PA security through the indistinguishability of two worlds. In the first and second worlds, an adversary can polynomially many times access to the decryption oracle and the extractor respectively. We say that a PKE is *perfectly/statistically/computationally PA secure*, if these two worlds are perfectly/statistically/computationally indistinguishable for the adversary.

1.2 Our Contributions

Motivation. In order to see the motivation of our work, we review the intuition behind the PA-ness. Recall that the intuition behind the PA-ness is "$\mathcal{A}$ knows the decrypted plaintext M," and this intuition is realized by the existence of an extractor $\mathcal{K}$ which can extract M.

In the definition of the standard model PA-ness [BP04], an extractor $\mathcal{K}$ requires to extract polynomially many plaintexts $M_1, \ldots, M_n$. This means that the standard model PA-ness [BP04] requires that "$\mathcal{A}$ knows all of $M_1, \ldots, M_n$."

However, our intuition suggests that "$\mathcal{A}$ knows all of $M_1, \ldots, M_n$" holds if and only if all of the following facts holds: "$\mathcal{A}$ knows M_1,"..., and "$\mathcal{A}$ knows M_n." Therefore, the extractor $\mathcal{K}$ should be "decomposed" into the extractors $\mathcal{K}_1, \ldots, \mathcal{K}_n$. Here $\mathcal{K}_i$ is an extractor which can extract M_i.

We would like to know whether this intuition is true or not. Recall that the overwhelming-based PA-ness requires an extractor to extract only one plaintext. Therefore, if the above intuition is true, the extractor $\mathcal{K}$ for the standard model PA-ness of [BP04] can "decompose" into the extractors $\mathcal{K}_1, \ldots, \mathcal{K}_n$ of the overwhelming-based PA security. So, the above motivation can rephrase as follows: "Can we define the standard model PA-ness by using the overwhelming-based methodology?"

Two Approaches are Almost Equivalent. In this paper, we define *OverWhelming-Based PA security (OWB-PA) in the standard model* and study the relationship between the OWB-PA security and the indistinguishability-based PA security

[BP04]. In particular, we show that the extractor $\mathcal{K}$ for statistical PA-ness, which extracts $M_1, \ldots, M_n$ can be constructed from the extractor $\mathcal{K}_1, \ldots, \mathcal{K}_n$ of the OWB-PA security. Here $\mathcal{K}_i$ is an extractor which extracts M_i.

A naive definition of the OWB-PA security is obtained by "directly standard modelizing" the overwhelming-based PA security [BR94,BDPR98] of the random oracle model. However, we can show that the naive OWB-PA security seems to be equivalent to none of the perfect/statistical/computational PA security [BP04]. Therefore, we somewhat modify the definition of the OWB-PA security, assume a very weak condition on a PKE and show that this (modified) OWB-PA security is equivalent to the statistical PA-security under this condition.

The modification we use is allowing an adversary to access the decryption oracle, and giving an auxiliary input to the adversary. Our condition for a PKE is about secret keys. Recall that, in some PKE such as the Cramer-Shoup scheme [CS98], one public key has two or more corresponding secret keys. Our condition, named sk-*non-redundancy*, is as follows: "If two secret keys sk_1 and sk_2 correspond to the same public key, $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C)$ holds for any ciphertext C." Clearly, this condition is satisfied for any honestly generated ciphertext $C = \mathsf{Enc}_{\mathsf{pk}}(M)$, because $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C) = M$ holds. The heart of the sk-non-redundancy is that $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C)$ holds even for maliciously generated ciphertext C. We can say that our sk-non-redundancy condition is very weak, because all known PKEs satisfy this condition.

Significance. One of the most significant point of our result is that it shows the "independence" of knowledge extractions. Recall that our result shows that the extractor $\mathcal{K}$ for the statistical PA-ness can be "decomposed" into the extractor $\mathcal{K}_1, \ldots, \mathcal{K}_n$ of the OWB-PA security. Here $\mathcal{K}$ is an extractor which extracts all $M_1, \ldots, M_n$ from decryption queries $C_1, \ldots, C_n$ of an adversary and $\mathcal{K}_i$ is an extractor which extracts M_i from C_i. Since $\mathcal{K}_i$ can extract M_i independent from other $\mathcal{K}_j$, this means that the knowledge extractions of M_i and M_j are "independent" from each other.

This independence is non-trivial fact from the folloing reason. Recall that the definition of the statistical PA-ness requires that $(M_1, \ldots, M_n) \simeq (\mathsf{Dec}_{\mathsf{sk}}(C_1), \ldots, \mathsf{Dec}_{\mathsf{sk}}(C_n))$ holds. Here "$\simeq$" denote the statistical indistinguishability.

However, the statistical indistinguishability $(X_1, \ldots, X_n) \simeq (Y_1, \ldots, Y_n)$ may not hold even if $X_1 \simeq Y_1, \ldots, X_n \simeq Y_n$ holds, where X_i and Y_i are random variables. (In fact, $(X_1, \ldots, X_n) \simeq (Y_1, \ldots, Y_n)$ hold only if the distribution of $X_1, \ldots, X_n$ are independent from each other.) Recall that an adversary of the statistical PA-ness can output $\{C_i\}_i$ such that the distribution of C_i is *not* independent from that of other C_j. Therefore, if $\mathcal{K}$ extracts $M_i \simeq \mathsf{Dec}_{\mathsf{sk}}(C_i)$ one by one, $(M_1, \ldots, M_n) \simeq (\mathsf{Dec}_{\mathsf{sk}}(C_1), \ldots, \mathsf{Dec}_{\mathsf{sk}}(C_n))$ may not holds.

Our result is non-trivial because it shows that $(M_1, \ldots, M_n) \simeq (\mathsf{Dec}_{\mathsf{sk}}(C_1), \ldots, \mathsf{Dec}_{\mathsf{sk}}(C_n))$ always holds even if $\mathcal{K}$ extracts $M_i \simeq \mathsf{Dec}_{\mathsf{sk}}(C_i)$ one by one by using the extractor K_i of the OWB-PA-ness. That is, our result shows that the "independence" of knowledge extraction holds even if the distributions of $C_1, \ldots, C_n$ are dependent.

More Detailed Studies about the Equivalence. As mentioned before, we show that the OWB-PA security was equivalent to the statistical PA security [BP04] only if a PKE is sk-non-redundant. However, we also consider a slightly modified version of the PA security [BP04] (named sk-*PA security*), where a distinguisher is provided with the secret key. Then we show that the OWB-PA security is equivalent to the sk-statistical PA security, even if a PKE is not sk-non-redundant.

In the statistical case, we can say that the difference between the sk-PA security and the original PA security is quite small, because we can show that these two notions are equivalent for a sk-non-redundant PKE and all known PKEs are sk-non-redundant.

However, the definition of the computational PA security dramatically changes if a distinguisher is provided with the secret key. In fact, we can prove that the sk-computational PA security is equivalent to the sk-statistical PA security, although the original computational PA security is strictly weaker than the original statistical PA security [TO06,TO08].

We can say that the above result show what the difference between the computational PA security and the statistical PA is. That is, we can say that the only difference between the computational PA security and the statistical PA security is in the knowledge of sk.

Computational PA-ness. We finally note about the computational PA-ness. One may think that our result can be generalized to the case of the computational PA-ness. That is, one may think that the computational PA-ness is equivalent to the "computational OWB-PA-ness." Here the computational OWB-PA-ness is a variant of the OWB-PA-ness such that an extractor requires to extract a plaintext only from one ciphertext and the extracted plaintext is only required to be computationally indistinguishable from the decrypted plaintext.

However, Bellare and Palacio [BP04] already showed that such computational OWB-PA-ness was strictly weaker than the computational PA-ness. (They used the term "PA0-ness" for the computational OWB-PA-ness.)

2 Standard Model PA-ness

We review the definition of the standard model PA-ness, which was given by Bellare and Palacio [BP04] and was given through indistinguishability-based methodology. From a technical reason, we slightly change the definition of [BP04]. That is,

- we give an auxiliary input to an adversary.

We will see in Subsection 4.2 why we need this modification.

Definition 1 (Standard Model PA-ness[BP04]) Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE. Let $\mathcal{A}$, $\mathcal{K}$, $\mathcal{P}$ be polytime machines, which are respectively called *adversary*, *extractor*, and *plaintext creator*.

$\text{—}\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z)\text{—}$	$\text{—}\mathsf{PA}^{\mathcal{K}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z)\text{—}$
Take random tapes R and μ for $\mathcal{A}$ and $\mathcal{P}$.	Take random tapes R, μ, and ρ for $\mathcal{A}$, $\mathcal{P}$, $\mathcal{K}$.
$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.	$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
	Initialize the list EList to ε.
	Initialize the state $\mathsf{St}_{\mathcal{K}}$ of $\mathcal{K}$ to ε.
Run $\mathcal{A}(\mathsf{pk},z;R)$ until it halts:	Run $\mathcal{A}(\mathsf{pk},z;R)$ until it halts:
If $\mathcal{A}$ makes an encryption query (enc,Q)	If $\mathcal{A}$ makes an encryption query (enc,Q)
$C \leftarrow \mathsf{Enc}_{\mathsf{pk}} \circ \mathcal{P}(Q;\mu)$.	$C \leftarrow \mathsf{Enc}_{\mathsf{pk}} \circ \mathcal{P}(Q;\mu)$, $\mathsf{EList} \leftarrow \mathsf{EList} \Vert C$.
Send C to $\mathcal{A}$ as the reply.	Send C to $\mathcal{A}$ as the reply.
If $\mathcal{A}$ makes a decryption query (dec,Q)	If $\mathcal{A}$ makes a decryption query (dec,Q)
$M \leftarrow \mathsf{Dec}_{\mathsf{sk}}(Q)$.	$(M,\mathsf{St}_{\mathcal{K}}) \leftarrow \mathcal{K}(\mathsf{pk},z,Q,R,\mathsf{EList},\mathsf{St}_{\mathcal{K}};\rho)$.
Send M to $\mathcal{A}$ as the reply.	Send M to $\mathcal{A}$ as the reply.
Return an output T of $\mathcal{A}$.	Return an output T of $\mathcal{A}$.

Fig. 1. Experiments for the Standard Model PA-ness of Bellare-Palacio [BP04]

For a plaintext creator $\mathcal{P}$, let $\mathsf{St}_{\mathcal{P}}$ and μ denote the state of $\mathcal{P}$ and the random tape of $\mathcal{P}$ respectively. The state $\mathsf{St}_{\mathcal{P}}$ is initialized to the null string ε. We let $\mathsf{Enc}_{\mathsf{pk}} \circ \mathcal{P}(Q;\mu)$ denote the algorithm which executes the following procedures: $(M,\mathsf{St}_{\mathcal{P}}) \leftarrow \mathcal{P}(Q,\mathsf{St}_{\mathcal{P}};\mu)$, $C \leftarrow \mathsf{Enc}_{\mathsf{pk}}(M)$, and output C.

For a security parameter λ, a polynomial poly, and an auxiliary input $z \in \{0,1\}^{\mathsf{poly}(\lambda)}$ of $\mathcal{A}$, we define two experiments $\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z)$ and $\mathsf{PA}^{\mathcal{K}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z)$, shown in Fig. 1. For a distinguisher $\mathcal{D}$, we set

$$P_{\mathcal{A},\mathsf{poly},\mathcal{K},\mathcal{P},\mathcal{D}}(\lambda) = \max_{z\in\{0,1\}^{\mathsf{poly}(\lambda)}} |\Pr[\mathcal{D}(\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z))=1]-\Pr[\mathcal{D}(\mathsf{PA}^{\mathcal{K}}_{\Pi,\mathcal{A},\mathsf{Enc}\circ\mathcal{P}}(\lambda,z))=1]|.$$

We say that a PKE Π is *perfectly, statistically, or computationally PA secure (with auxiliary input)* if it satisfies the following property 1, 2, or 3 respectively.

1. ${}^\forall\mathcal{A}{}^\forall\mathsf{poly}{}^\exists\mathcal{K}{}^\forall\mathcal{P}{}^\forall\mathcal{D}$ (superpolytime distinguisher)${}^\forall\lambda$: $P_{\mathcal{A},\mathsf{poly},\mathcal{K},\mathcal{P},\mathcal{D}}(\lambda) = 0$.
2. ${}^\forall\mathcal{A}{}^\forall\mathsf{poly}{}^\exists\mathcal{K}{}^\forall\mathcal{P}{}^\forall\mathcal{D}$ (superpolytime distinguisher) : $P_{\mathcal{A},\mathsf{poly},\mathcal{K},\mathcal{P},\mathcal{D}}(\lambda)$ is negligible for λ.
3. ${}^\forall\mathcal{A}{}^\forall\mathsf{poly}{}^\exists\mathcal{K}{}^\forall\mathcal{P}{}^\forall\mathcal{D}$ (polytime distinguisher) : $P_{\mathcal{A},\mathsf{poly},\mathcal{K},\mathcal{P},\mathcal{D}}(\lambda)$ is negligible for λ.

We say that $\mathcal{K}$ is *successful* for $\mathcal{A}$ if it satisfies the above relation for any $\mathcal{P}$ and any $\mathcal{D}$.

We stress that $(\mathsf{pk},\mathsf{sk})$ *is chosen after z is determined.* This fact is important. In fact, if the auxiliary input z depends on $(\mathsf{pk},\mathsf{sk})$, the definition of the PA-ness become meaningless. If we allow z to depend on $(\mathsf{pk},\mathsf{sk})$, z can be equal to some ciphertext $z = \mathsf{Enc}_{\mathsf{pk}}(M)$. Then $\mathcal{A}$ can obtain an auxiliary input $z = \mathsf{Enc}_{\mathsf{pk}}(M)$ "without knowing" the plaintext M. Then clearly, no extractor can obtain M, if $\mathsf{Enc}_{\mathsf{pk}}$ is oneway. Therefore, no non-trivial scheme satisfies the PA-ness.

3 Definition of Overwhelming-Based Standard Model PA

3.1 Definition

We review the definition of the random oracle PA-ness [BR94,BDPR98], because the random oracle PA-ness is given through the overwhelming-based approach.

```
Take random tapes $R$ and $\rho$ for $\mathcal{A}$ and $\mathcal{K}$.
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}^{\mathsf{Hash}}(1^{\lambda})$.
$C_0 \leftarrow \mathcal{A}^{\mathsf{Hash}, \mathsf{Enc}_{\mathsf{pk}}^{\mathsf{Hash}}}(\mathsf{pk}; R)$.
$\mathsf{EList} \leftarrow$(The list of all answers from the oracle $\mathsf{Enc}_{\mathsf{pk}}^{\mathsf{Hash}}$).
$\mathsf{HList} \leftarrow$(The list of all pairs of hash queries of $\mathcal{A}$ and the corresponding answers).
$M_0 \leftarrow \mathcal{K}(\mathsf{pk}, C_0, \mathsf{EList}, \mathsf{HList}; \rho)$.
If $M_0 = \mathsf{Dec}_{\mathsf{sk}}^{\mathsf{Hash}}(C_0)$ return 1. Otherwise return 0.
```

Fig. 2. Experiment used to define the random oracle PA security [BDPR98]

Definition 2 (Overwhelming-Based PA Security in the Random Oracle Model [BR94,BDPR98]) Let $\Pi = (\mathsf{Gen}^{\mathsf{Hash}}, \mathsf{Enc}^{\mathsf{Hash}}, \mathsf{Dec}^{\mathsf{Hash}})$ be a PKE which uses a hash function Hash. Let $\mathcal{A}$ and $\mathcal{K}$ be polytime machines, which are respectively called *adversary* and *extractor*. For a security parameter λ, we define an experiment $\mathsf{OWB\text{-}PA}^{\mathsf{RO}}_{\Pi,\mathcal{A},\mathcal{K},\mathsf{Enc}}(\lambda)$ as in Fig.2. In this experiment, C_0 must not be an element of EList.

We say that Π is *OverWhelming-Based PA secure (OWB-PA) in the random oracle model*, if Π satisfies the following property:

$$^{\exists}\mathcal{K}^{\forall}\mathcal{A} : \Pr[\mathsf{OWB\text{-}PA}^{\mathsf{RO}}_{\Pi,\mathcal{A},\mathcal{K},\mathsf{Enc}}(\lambda) \neq 1] \text{ is negligible for } \lambda.$$

We give an overwhelming-based standard model PA-ness by modifying the above definition in the following ways:

1. We "directly standard modelize" Definition 2. That is,
   (a) We remove the random oracle.
   (b) We allow a non-black-box extractor.
   (c) We add a plaintext creator $\mathcal{P}$.
2. We give an auxiliary input to $\mathcal{A}$.
3. We allow an adversary to access the decryption oracle.

As mentioned in [BP04], the modifications (a), (b), and (c) are definitely required when we define the standard model PA-ness. The modification 2 and 3 are required in order to show the equivalence between the OWB-PA-ness and the indistinguishability-based statistical PA-ness. See Subsection 4.2 for the details.

Definition 3 (OverWhelming-Based PA security (OWB-PA) in the Standard Model) We take $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, $\mathcal{A}$, $\mathcal{K}$, $\mathcal{P}$, λ, and poly, as in Definition 1. We let define $\mathsf{Enc}_{\mathsf{pk}} \circ \mathcal{P}$ as in the Definition 1. For an auxiliary input $z \in \{0,1\}^{\mathsf{poly}(\lambda)}$ of $\mathcal{A}$, we define an experiment $\mathsf{OWB\text{-}PA}_{\Pi,\mathcal{A},\mathcal{K},\mathsf{Enc}\circ\mathcal{P}}(\lambda, z)$ as in Fig.3. In this experiment, C_0 must not be an element of EList.

We say that Π satisfies *OverWhelming-Based PA security (OWB-PA) in the standard model*, if it satisfies the following property:

$$^{\forall}\mathcal{A}^{\forall}\mathsf{poly}^{\exists}\mathcal{K}^{\forall}\mathcal{P} : \max_{z \in \{0,1\}^{\mathsf{poly}(\lambda)}} \Pr[\mathsf{OWB\text{-}PA}_{\Pi,\mathcal{A},\mathcal{K},\mathsf{Enc}\circ\mathcal{P}}(\lambda, z) \neq 1] \text{ is negligible for } \lambda.$$

We say that $\mathcal{K}$ is *successful* for $\mathcal{A}$ if it satisfies the above property for any $\mathcal{P}$.

```
—$\mathsf{OWB\text{-}PA}_{\Pi,\mathcal{A},\mathcal{K},\mathsf{Enc}\circ\mathcal{P}}(\lambda, z)$—
Take random tapes $R$, $\rho$, and $\mu$ for $\mathcal{A}$, $\mathcal{K}$, and $\mathcal{P}$.
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
$C_0 \leftarrow \mathcal{A}^{\mathsf{Enc}_{\mathsf{pk}}\circ\mathcal{P}(\cdot;\mu),\mathsf{Dec}_{\mathsf{sk}}}(\mathsf{pk}, z; R)$
$\mathsf{EList} \leftarrow$(The list of all answers from the oracle $\mathsf{Enc}_{\mathsf{pk}}$).
$\mathsf{DList} \leftarrow$(The list of all answers from the oracle $\mathsf{Dec}_{\mathsf{sk}}$).
$M_0 \leftarrow \mathcal{K}(\mathsf{pk}, z, C_0, R, \mathsf{EList}, \mathsf{DList}; \rho)$.
If $M_0 = \mathsf{Dec}_{\mathsf{sk}}(C_0)$, return 1. Otherwise return 0.
```

Fig. 3. Experiment used to define the Definition of OWB-PA security

3.2 The Decryption Oracle Strengthens the Definition

At first glance, the modification 3 of Subsection 3.1 seems to be meaningless, because (1) the OWB-PA security (with or without the modification 3) means that "an adversary $\mathcal{A}$ knows a plaintext corresponding to the ciphertext generated by $\mathcal{A}$," (2) in particular, "an adversary knows the plaintext M_i corresponding to the i-th decryption query C_i," (3) therefore, an adversary can obtain M_i without accessing the decryption oracle.

However, the above discussion is not true. Recall that the intuition "an adversary $\mathcal{A}$ knows a plaintext" is realized by using a polytime extractor. Therefore, "an adversary knows the plaintext M_i corresponding to the i-th decryption query C_i" means that "there exists a polytime extractor $\mathcal{K}_i$ which can extract M_i from C_i." The problem is in the dependency of $\mathcal{K}_i$ on i. Suppose that $\mathcal{A}$ makes decryption query λ times, where λ is the security parameter. Since $\mathcal{K}_i$ depends on i, the number of steps T_i of $\mathcal{K}_i$ also depends on i. Therefore, it is possible that $T_i = 2^i p_i(\lambda)$ holds for some polynomial p_i.

For each fixed i, the number of steps $T_i = 2^i p_i(\lambda)$ of $\mathcal{K}_i$ is polynomial of the security parameter λ. Therefore, each $\mathcal{K}_i$ is a polytime machine. However, $\mathcal{A}$ needs superpolytime if $\mathcal{A}$ executes all of $\mathcal{K}_1, \ldots, \mathcal{K}_\lambda$. Therefore, if $\mathcal{A}$ cannot access the decryption oracle, $\mathcal{A}$ needs superpolytime in order to obtain all of $M_1, \ldots, M_\lambda$. This means that the polytime adversary $\mathcal{A}$ cannot obtain all of $M_1, \ldots, M_\lambda$. Therefore, we can say that the decryption oracle is meaningful. Note that Bellare and Palacio [BP04] use similar discussions in other context.

4 OWB-PA Security Implies Statistical PA

4.1 Result

In this section, we prove that the OWB-PA-ness implies the statistical PA-ness:

Theorem 4 (OWB-PA ⇒ Statistical PA). *Let Π be a PKE satisfying the OWB-PA security. Then Π satisfies the statistical PA security.*

We here give the idea behind the proof. The formal proof will be depicted in the full paper.

Proof. (idea) Let $\mathit{\Pi}$ be an OWB-PA secure PKE, $\mathcal{A}_0$ be an adversary for the statistical PA-ness of $\mathit{\Pi}$ and n_0 be the number of decryption queries of $\mathcal{A}_0$. Bellow, z is an auxiliary input of $\mathcal{A}_0$ and $(\mathsf{pk}, \mathsf{sk})$ is a public key/secret key pair.

1. We construct an adversary $\mathcal{B}_0$ of the OWB-PA security such that, on input $(\mathsf{pk}, \mathtt{1}^i \| z)$, $\mathcal{B}_0$ outputs the i-th decryption query of $\mathcal{A}_0(\mathsf{pk}, z)$. The description of $\mathcal{B}_0(\mathsf{pk}, z')$ is as follows:
   - $\mathcal{B}_0$ parses z' as $\mathtt{1}^i \| z$. (If z' is not this type, $\mathcal{B}_0$ outputs $\perp$ and terminates.)
   - $\mathcal{B}_0$ executes $\mathcal{A}_0(\mathsf{pk}, z)$ if $i \leq n_0$. (Otherwise, $\mathcal{B}_0$ outputs $\perp$ and terminates.)
   - If $\mathcal{A}$ makes encryption queries $\mathcal{B}_0$ answers them by passing the queries to the encryption oracle of $\mathcal{B}_0$.
   - If $\mathcal{A}$ makes the j-th decryption query C_j for $j < i$, $\mathcal{B}_0$ answers them by passing the query to the decryption oracle of $\mathcal{B}_0$.
   - If $\mathcal{A}$ makes the i-th decryption query C_i, $\mathcal{B}_0$ outputs it and terminates.
2. From the OWB-PA security of $\mathit{\Pi}$, there exists an extractor $\mathcal{L}_0$ for $\mathcal{B}_0$.
3. We let $\mathcal{K}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$ be the algorithm which executes $\mathcal{L}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$, obtains an output M_i of $\mathcal{L}_0$, and outputs M_i.

Since $\mathcal{K}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$ executes the extractor $\mathcal{L}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$ for $\mathcal{B}_0(\mathsf{pk}, \mathtt{1}^i \| z)$, and since $\mathcal{B}_0$ outputs the i-th decryption query of $\mathcal{A}_0(\mathsf{pk}, \mathtt{1}^i \| z)$, the outputs M_i of $\mathcal{K}_0$ is equal to $\mathsf{Dec}_{\mathsf{sk}}(C_i)$ with overwhelming probability.

We show that the number T of steps of $\mathcal{K}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$ is bounded by some polynomial, which is independent from i. Note that the independency from i is quite important. If T depends on i, $T = 2^i p_i(\lambda)$ can hold for some polynomial $p_i(\lambda)$. This means that T become superpolynomial $T = 2^\lambda p_\lambda(\lambda)$ when $\mathcal{K}_0$ extracts a plaintext from λ-th decryption query of $\mathcal{A}$.

Since $\mathcal{K}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho) = \mathcal{L}_0(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$, we have to show the following facts in order to show that $\mathcal{K}_0$ is a polytime machine:

- The description of $\mathcal{L}_0$ is independent from i.
- The length of the input $(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}; \rho)$ of $\mathcal{L}_0$ is bounded by some polynomial, which is independent from i.

We can prove that the description of $\mathcal{L}_0$ is independent from i, because the $\mathcal{L}_0$ depends only on $\mathcal{B}_0$ and because the description of $\mathcal{B}_0$ is independent from i. We next prove that the length of the input $(\mathsf{pk}, \mathtt{1}^i \| z, C_i, R, \mathsf{EList}, \mathsf{St}, \rho)$ is bounded by some polynomial, which is independent from i. Recall that i is the number of decryption queries of $\mathcal{A}$. Since $\mathcal{A}$ is a polytime machine, this means that i is bounded by the polynomial n_0 which is independent from i. Here n_0 is the number of steps of $\mathcal{A}$. This means that the length of $\mathtt{1}^i$ is bounded by the polynomial n_0 which is independent from i. Moreover, from the definition of the statistical PA-ness, the length of z is bounded by some polynomial $\mathsf{poly}(\lambda)$, which is independent from i. The lengths of other inputs are clearly bounded by a polynomial which is independent from i. □

4.2 Why Are the Modified Definitions Required?

When we define the (standard model) OWB-PA-ness, we modify the random oracle OWB-PA-ness in two ways. That is, we give an auxiliary input to an adversary and allows an adversary to access the decryption oracle. Similarly, we slightly modify the original definition of the statistical PA-ness [BP04] and give an auxiliary input to an adversary for it.

We think that these modifications are quite important to show Theorem 4. In this subsection, we see why these modifications are required.

Effect of Auxiliary Inputs: In the proof of Subsection 4.1, we use an adversary $\mathcal{B}_0$ such that, by giving an auxiliary input $1^i\|z$, $\mathcal{B}_0$ outputs the i-th decryption query C_i of $\mathcal{A}_0$. Therefore, if we do not give adversaries to auxiliary inputs, we cannot use the proof of Subsection 4.1.

One way to "prove" Theorem 4 without using auxiliary inputs is to construct adversary $\mathcal{B}_i$ which depends on i. That is, we "prove" Theorem 4 as follows. Here $\mathcal{A}_0$ is an adversary for the statistical PA security. We would like to construct an extractor for $\mathcal{A}_0$.

- For each i, we construct an adversary $\mathcal{B}_i$ for the OWB-PA security, such that $\mathcal{B}_i$ outputs the i-th decryption query C_i of $\mathcal{A}_0$. (Contrary to the previous $\mathcal{B}_0$, each i is coded in the program of $\mathcal{B}_i$. Therefore, $\mathcal{B}_i$ does not require an auxiliary input $1^i\|z$.)
- From the OWB-PA-ness of the PKE Π, there exists extractor $\mathcal{L}_i$ for each $\mathcal{B}_i$.
- We construct an extractor $\mathcal{K}_0$ for $\mathcal{A}_0$ such that $\mathcal{K}_0$ uses $\mathcal{L}_i$ in order to extract a plaintext from C_i.

The failure of the above "proof" is that the above $\mathcal{K}_0$ may be superpolytime machine. The reason is as follows. In the above "proof," we construct $\mathcal{B}_i$ which depends on i. Hence, the extractor $\mathcal{L}_i$ of $\mathcal{B}_i$ depends on i also. Therefore, the number T_i of steps of $\mathcal{L}_i$ can depend on i. Therefore, it is possible that $T_i = 2^i p_i(\lambda)$ holds for some polynomial p_i.

For each fixed i, the number of steps $T_i = 2^i p_i(\lambda)$ of $\mathcal{L}_i$ is polynomial of the security parameter λ. Therefore, $\mathcal{L}_i$ is a polytime extractor of $\mathcal{B}_i$ for the OWB-PA security. However, $\mathcal{K}_0$ becomes a superpolynomial extractor, because $\mathcal{K}_0$ uses all of $\mathcal{L}_1, \ldots, \mathcal{L}_{n_0}$ and therefore requires steps more than $2^{n_0} p_{n_0}(\lambda)$. Here n_0 is the number of steps of $\mathcal{A}_0$ and therefore is a polynomial of λ.

Effect of the Decryption Oracle: In the proof of Subsection 4.1, we use an adversary $\mathcal{B}_0$ which accesses the decryption oracle. Therefore, if we do not allow an adversary to access the decryption oracle, we cannot use the proof of Subsection 4.1.

One way to to "prove" Theorem 4 without using the decryption oracle is to construct adversaries and their extractors recursively. That is, we seem to "prove" Theorem 4 as follows. Here $\mathcal{A}_0$ is an adversary for the statistical PA security. We would like to construct an extractor for $\mathcal{A}_0$.

– For each i, we construct an adversary $\mathcal{B}_i$ for the OWB-PA-ness and its extractor $\mathcal{L}_i$ recursively:
  - We define $\mathcal{B}_i$ as follows: $\mathcal{B}_i$ executes $\mathcal{A}_0$ and answers the j-th decryption query C_j of $\mathcal{A}_0$ by using $\mathcal{L}_j$ for $j < i$, and outputs i-th decryption query C_i of $\mathcal{A}_0$.
  - We set $\mathcal{L}_i$ to an extractor of $\mathcal{B}_i$ for the OWB-PA-ness.

– We construct an extractor $\mathcal{K}_0$ for $\mathcal{A}_0$ such that $\mathcal{K}_0$ uses $\mathcal{L}_i$ in order to extract a plaintext from C_i.

The failure of the above "proof" is that the above $\mathcal{K}_0$ may be superpolytime machine. The reason is similar to that for an auxiliary input. In the above "proof," $\mathcal{B}_i$ and $\mathcal{L}_i$ depends on i also. Therefore, it is possible that the number T_i of steps of $\mathcal{L}_i$ satisfies $T_i = 2^i p_i(\lambda)$ for some polynomial p_i.

For each fixed i, the number of steps $T_i = 2^i p_i(\lambda)$ of $\mathcal{L}_i$ is polynomial of the security parameter λ. Therefore, $\mathcal{L}_i$ is a polytime extractor of $\mathcal{B}_i$ for the OWB-PA security. However, $\mathcal{K}_0$ becomes a superpolynomial extractor, because $\mathcal{K}_0$ uses all of $\mathcal{L}_1, \ldots, \mathcal{L}_{n_0}$ and therefore requires steps more than $2^{n_0} p_{n_0}(\lambda)$. Here n_0 is the number of steps of $\mathcal{A}_0$ and therefore is a polynomial of λ.

5 The Statistical PA Is Equivalent to the OWB-PA Security, Under Very Weak Condition

We already showed that the OWB-PA security implied the statistical PA security of Section 2. In this section, we show that the converse holds under very weak condition.

5.1 Equivalency under very weak condition

We first give the condition (named sk-*non-redundancy*), under which the OWB-PA security is equivalent to the statistical PA security. Recall that each public key pk of a some PKE, such as the Cramer-Shoup scheme [CS98,CS01], has many corresponding secret keys. (Here we say that a public key pk *corresponds* to sk, if there exists a random tape ν satisfying $(\mathsf{pk}, \mathsf{sk}) = \mathsf{Gen}(1^\lambda; \nu)$.) Intuitively, the sk-non-redundancy is the condition which ensures that $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C)$ holds with overwhelming probability for any secret keys sk_1 and sk_2 corresponding to the same public key pk. Clearly, this condition is satisfied for any honestly generated ciphertext $C = \mathsf{Enc}_{\mathsf{pk}}(M)$, because $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C) = M$ holds. The heart of the sk-non-redundancy is that $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_2}(C)$ holds even for maliciously generated ciphertext C.

We can say that our sk-non-redundancy condition is very weak, because all known PKEs satisfy this condition. However, we can give an artificial example $\Pi' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ of Fig.4 such that Π' is not sk-non-redundant. Here $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is an arbitrary PKE. Since $\mathsf{sk}' = \mathsf{sk} \| R$ holds and since $\mathsf{Dec}_{\mathsf{sk}'}(1\|C)$ is equal to R, the output $\mathsf{Dec}_{\mathsf{sk}'}(1\|C)$ varies depending on a secret

```
—$\mathsf{Gen}'(1^\lambda)$—
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$
$R \leftarrow$ ($\lambda$-bit random bit string).
$\mathsf{pk}' \leftarrow \mathsf{pk}$, $\mathsf{sk}' \leftarrow \mathsf{sk} \| R$.
Output $(\mathsf{pk}', \mathsf{sk}')$.
—$\mathsf{Enc}'_{\mathsf{pk}'}(M)$—
$C \leftarrow \mathsf{Enc}_{\mathsf{pk}}(M)$, $C' \leftarrow 0 \| C$. Output $C'$.
—$\mathsf{Dec}'_{\mathsf{sk}'}(C')$—
Parse $C'$ as $b \| C$.
If $b = 0$, output $\mathsf{Dec}_{\mathsf{sk}}(C)$.
Otherwise, output $R$.
```

Fig. 4. A Scheme Π' which is not sk-non-redundant

key sk', even if the corresponding public key pk' does not vary. Note that Bellare and Palacio [BP04] used a similar scheme in other context.

We now formalize the sk-non-redundancy. Recall that the sk-non-redundancy means that $\mathsf{Dec}_{\mathsf{sk}_1}(C) = \mathsf{Dec}_{\mathsf{sk}_1}(C)$ holds for any secret keys sk_1 and sk_2 corresponding to the same public key pk. In other words, $\mathsf{Dec}_{\mathsf{sk}}(C)$ depends only on pk and C, and therefore does not depend on sk. If $\mathsf{Dec}_{\mathsf{sk}}(C)$ is determined from pk and C, we can define a (superpolytime) function $\overline{\mathsf{Dec}}$ satisfying $\overline{\mathsf{Dec}}_{\mathsf{pk}}(C) = \mathsf{Dec}_{\mathsf{sk}}(C)$.

Definition 5 Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE. We say that Π satisfies sk-*non-redundancy* if there exists a *superpolytime* deterministic function $\overline{\mathsf{Dec}}$ such that

$$\max_{\substack{C \in \{0,1\}^* \\ \mathsf{pk}_0 \in \{0,1\}^*}} \Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda) : \mathsf{Dec}_{\mathsf{sk}}(C) \neq \overline{\mathsf{Dec}}_{\mathsf{pk}_0}(C) \mid \mathsf{pk} = \mathsf{pk}_0] \text{ is negligible for } \lambda.$$

We next give our main result:

Theorem 6 (OWB-PA = Statistical PA under sk-non-redundancy). *Let Π be a sk-non-redundant PKE. Then Π is statistically PA secure if and only if OWB-PA secure.*

The "only-if" part of the above theorem has already been shown in Theorem 4. We give the idea behind the proof of the "if"-part. The formal proof will be described in the full paper.

Proof. (idea) Let Π be a PKE which is sk-non-redundant and is statistically PA secure. Let $\mathcal{A}_0$ be an adversary for the OWB-PA security, $(\mathsf{pk}, \mathsf{sk})$ be a public key/secret key pair and z is an auxiliary input for $\mathcal{A}_0$. We construct an adversary $\mathcal{B}_0$ for the statistical PA security as follows. $\mathcal{B}_0(\mathsf{pk}, z)$ executes $\mathcal{A}_0(\mathsf{pk}, z)$. If $\mathcal{A}_0$ makes a decryption query, $\mathcal{B}_0$ answers it by passing it to the decryption oracle. $\mathcal{A}_0$ finally outputs a ciphertext C_0 and terminates. Then $\mathcal{B}_0$ makes decryption query

C_0, obtains answer M_0 to the query, outputs (pk, C_0, M_0), and terminates. From the assumption, there is an extractor $\mathcal{K}_0$ for $\mathcal{B}_0$ of the statistical PA security.

We construct a superpolytime distinguisher $\mathcal{D}_0$ which tries to distinguish an output of $\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{B}_0,\mathsf{Enc}\circ\mathcal{P}_0}(\lambda, z)$ and that of $\mathsf{PA}^{\mathcal{K}_0}_{\Pi,\mathcal{B}_0,\mathsf{Enc}\circ\mathcal{P}_0}(\lambda, z)$, where $\mathcal{P}_0$ is a plaintext creator. $\mathcal{D}_0(\mathsf{pk}, C_0)$ computes (one of) a secret key sk' corresponding to pk by using superpolytime. Then $\mathcal{D}_0$ outputs 1 or 0, depending on whether $M_0 = \mathsf{Dec}_{\mathsf{sk}'}(C_0)$ holds or not.

In $\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{B}_0,\mathsf{Enc}\circ\mathcal{P}_0}(\lambda, z)$, the decryption oracle sends the answer $\mathsf{Dec}_{\mathsf{sk}}(C_0)$ to $\mathcal{A}_0$. From the sk-non-redundancy, $\mathsf{Dec}_{\mathsf{sk}'}(C_0) = \mathsf{Dec}_{\mathsf{sk}}(C_0)$ holds with overwhelming probability. Therefore, $\mathcal{D}_0$ outputs 1 if (pk, C_0, M_0) is an output of $\mathsf{PA}^{\mathsf{Dec}}_{\Pi,\mathcal{B}_0,\mathsf{Enc}\circ\mathcal{P}_0}(\lambda, z)$. This means that even if (pk, C_0, M_0) is an output of $\mathsf{PA}^{\mathcal{K}_0}_{\Pi,\mathcal{B}_0,\mathsf{Enc}\circ\mathcal{P}_0}(\lambda, z)$, $\mathcal{D}_0$ outputs 1 with overwhelming probability. That is, an output of $\mathcal{K}_0$ is equal to $\mathsf{Dec}_{\mathsf{sk}}(C_0)$ with overwhelming probability. This means that $\mathcal{K}_0$ can use an extractor for $\mathcal{A}_0$ of the OWB-PA security. Since $\mathcal{A}_0$ is an arbitrary adversary for the OWB-PA security, this means that Π is OWB-PA secure.□

5.2 Effect of sk-non-redundancy

The sk-non-redundancy is important to show Theorem 6. In fact, we can show that the OWB-PA security does not imply the statistical PA security, if we suppose no assumption for the PKE:

Theorem 7 (Perfect, Statistical and Computational PA $\not\Rightarrow$ OWB-PA). *Suppose the existence of a perfectly (resp. statistically, computationally) PA secure PKE in the standard model. Then there exists a PKE which is not OWB-PA secure but is perfectly (resp. statistically, computationally) PA secure in the sence of Section 2.*

Proof. (idea) We only show the theorem for the case of the statistical PA security. We can show the theorem for other cases quite similarly.

Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE which is statistically PA secure. By using Π, we construct another PKE $\Pi' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ as in Fig.4. We show that Π' is not OWB-PA secure. Let $\mathcal{A}$ be an adversary which outputs $C' = 1\|\mathsf{Enc}_{\mathsf{pk}'}(0)$. Then an extractor $\mathcal{K}$ for $\mathcal{A}$ has to output $R = \mathsf{Dec}_{\mathsf{sk}'}(C')$. However, $\mathcal{K}$ succeeds in outputting R with only negligible probability, because the distribution of R is independent from the view of $\mathcal{K}$. This means that Π' is not OWB-PA secure.

We next show that Π' is statistically PA secure. Let $\mathcal{A}$ be an adversary for Π'. We can recognize $\mathcal{A}$ as adversary for Π. Since Π is statistical PA secure, there exists an extractor $\mathcal{K}$ of $\mathcal{A}$ for Π. We construct an extractor $\mathcal{K}'$ of $\mathcal{A}$ for Π' as follows. $\mathcal{K}'$ selects R' randomly and fixed it. If $\mathcal{K}'$ is provided with a ciphertext $C' = 0\|C$ for some C, $\mathcal{K}'$ executes $\mathcal{K}$ by giving C, obtains the output M of $\mathcal{K}$, and sends M back to $\mathcal{A}$. If $\mathcal{K}'$ is provided with a ciphertext $C' = 1\|C$ for some C, $\mathcal{K}'$ sends R' back to $\mathcal{A}$.

We see that $\mathcal{K}'$ is a successful extractor. Since $\mathcal{K}$ is a successful extractor, if $C' = 0\|C$ holds, $\mathcal{K}'$ obviously succeeds in simulating the decryption oracle with overwhelming probability. Since the distribution of R is independent from the view of $\mathcal{A}$, $\mathcal{A}$ cannot distinguish R and R'. Therefore, even if $C' = 1\|C$ holds, $\mathcal{K}'$ succeeds in simulating the decryption oracle with overwhelming probability.□

6 The sk-PA Security

We showed that the OWB-PA security was equivalent to the statistical PA security [BP04] only if a PKE was sk-non-redundant. In this section, we consider a slightly modified version of the PA security [BP04] (named sk-*PA security*), where a distinguisher is provided with the secret key. Then we see that the OWB-PA security is equivalent to the sk-statistical PA security, even if a PKE is not sk-non-redundant. The formal definition of the sk-PA security will depicted in the full paper. Note that Fujisaki [F06] also considered a variant of a PA-ness where a distinguisher is provided with the secret key.

The modification that we give the secret key to a distinguisher is quite small, in the case of statistical PA security. In fact, since a distinguisher $\mathcal{D}$ of the statistical PA security is a superpolytime machine, $\mathcal{D}$ can compute a secret key corresponding to the public key pk by using superpolytime. However, there may be many secret keys corresponding to pk as mentioned in Subsection 5.1, and $\mathcal{D}$ cannot know which one is true sk. Therefore, we can say that the only advantage of the sk-statistical PA security is that the distinguisher can know which one is sk.

If a PKE is sk-non-redundant, $\mathsf{Dec}_{\mathsf{sk}}(C) = \mathsf{Dec}_{\mathsf{sk}'}(C)$ holds for any sk and sk$'$ corresponding to the same public key pk. Therefore, the sk-statistical PA security is not advantageous to the statistical PA security, in this case. Hence, we can show the following theorem. The proof will be described in the full paper.

Theorem 8. (statistical PA = sk-statistical PA, under sk-non-redundancy) *Suppose that a PKE Π satisfies the* sk*-non-redundancy. Then Π satisfies the statistical PA security if and only if it satisfies the* sk*-statistical PA security.*

We now give our result.

Theorem 9. (OWB-PA = sk-statistical PA = sk-computational PA) *The following properties are equivalent:*

- *the OWB-PA security.*
- *the* sk*-statistical PA security.*
- *the* sk*-computational PA security.*

We can prove the above theorem in a similar way to that of Theorem 4. The proof will be described in the full paper. Note that we can generalize Theorem 9 into the case of the perfect PA security, if we allow an extractor to output fail with negligible probability.

One of the most surprising fact of the above theorem is that the sk-statistical PA security is equivalent to the sk-computational PA security. This fact is impressed because the statistical PA security is strictly stronger than the computational PA security [TO06,TO08]. Therefore we can say that the only difference between the statistical PA security and the computational PA security is in the knowledge of sk.

We can also define more stronger variant of PA security, named the *View-PA security*, such that a distinguisher is given the views of all entities. Above, "the views of all entities" means the key generation algorithm Gen, an adversary $\mathcal{A}$, a plaintext creator $\mathcal{P}$, and the encryption oracle $\mathsf{Enc}_{\mathsf{pk}}(\cdot)$. Then it is also equivalent to the OWB-PA security. We will describe the proof in the full paper.

Theorem 10 (OWB-PA = View-statistical PA = View-computational PA).
The following properties are equivalent:

- *the OWB-PA security.*
- *the View-statistical PA security.*
- *the View-computational PA security.*

7 Conclusion

There were two approaches to define the PA-ness, the indistinguishability-based approach and the overwhelming-based approach. The current definition [BP04] of the PA-ness was given by using the indistinguishability-based approach.

In this paper, we defined an alternative definition of the (standard model) PA-ness, OWB-PA security, based on the overwhelming-approach. Basically, this notion was given by "standard modelizing" the random oracle model PA-ness [BR94,BDPR98]. However, we essentially changed the definition in one point, that is, we allowed an adversary to access the decryption oracle.

We then showed that our OWB-PA security was equivalent to the statistical PA security of [BP04], under a very weak condition, the sk-non-redundancy. We also gave a new definition of the PA-ness, named sk-PA-ness, and showed that the OWB-PA security was equivalent to the sk-statistical PA-ness, even if a PKE was not sk-non-redundant.

References

[BDPR98] Mihir Bellare, Anand Desai, David Pointcheval, Phillip Rogaway. Relations Among Notions of Security for Public-Key Encryption Schemes. CRYPTO 1998. pp.26–45.

[BP04] Mihir Bellare and Adriana Palacio. Towards plaintext-aware public-key encryption without random oracles. ASIACRYPT 2004. pp. 48–62.

[BR94] Mihir Bellare, Phillip Rogaway. Optimal Asymmetric Encryption. EUROCRYPT 1994. pp.92–111.

[BD07] James Birkett and Alexander W. Dent. Relations Among Notions of Plaintext Awareness. PKC 2008, pp.47–64. eprint 2007/291.

[CS98] Ronald Cramer, Victor Shoup. A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack. CRYPTO 1998. pp.13–25.

[CS01] Ronald Cramer, Victor Shoup. Design and Analysis of Practical Public-Key Encryption Schemes. 2001.

[D91] Ivan Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In CRYPTO'91. pp.445–456.

[D06] Alexander W. Dent. Cramer-Shoup is Plaintext-Aware in the Standard Model. EUROCRYPT 2006.

[F06] Eiichiro Fujisaki. Plaintext Simulatability. IEICE Transactions 89-A(1), pp.55–65, 2006.

[FO99] Eiichiro Fujisaki, Tatsuaki Okamoto. How to Enhance the Security of Public-Key Encryption at Minimum Cost. PKC'99. pp. 53–68.

[HT06] Ryotaro Hayashi, Keisuke Tanaka. PA in the Two-Key Setting and a Generic Conversion for Encryption with Anonymity. ACISP 2006, pp.271–282

[HLM03] Jonathan Herzog, Moses Liskov, Silvio Micali. Plaintext Awareness via Key Registration. CRYPTO 2003, pp.548–564

[S01] Victor Shoup. OAEP Reconsidered. CRYPTO 2001, pp.239–259. J. Cryptology, 2002, 15(4), pp. 223–249.

[TO06] Isamu Teranishi, Wakawa Ogata. Relationship between Standard Model Plaintext Awareness and Message Hiding. ASIACRYPT 2006. pp. 226–240.

[TO08] Isamu Teranishi, Wakaha Ogata. The full paper of [TO06]. IEICE Transactions 91-A(1), pp.244–261, 2008.