

# 1 Preliminaries

## 1.1 Plaintext awareness

*Plaintext awareness (PA)* [BR95, BDPR98] describes the idea that no efficient algorithm can produce a valid ciphertext without knowing the corresponding decryption.

Let  $\text{PKE} = (\text{KeyGen}, \text{E}, \text{D})$  be a public-key encryption scheme. Let  $H$  be a hash function. Let  $\mathcal{E}_{\text{pk}}^H$  be an encryption oracle that takes no argument and returns valid ciphertexts when queried. Under the random oracle model, hash queries made to the oracle  $\mathcal{O}^H$  can be logged to a tape  $\mathcal{L}^H = \{(h_i, H(h_i))\}$ , and ciphertexts obtained from the encryption oracle are also logged to a separate tape  $C = \{c : c \leftarrow \mathcal{E}_{\text{pk}}^H(\cdot)\}$ .

A plaintext-awareness adversary  $B$  is a probabilistic algorithm that is given some public key and access to the two oracles, then output some ciphertext  $c$ . Note that the encryption oracle here is not redundant because obtaining ciphertexts from the encryption oracle will not log any corresponding hash queries in the hash oracle. This models a PA adversary's ability to obtain valid ciphertexts without running the encryption routine, such as by eavesdropping. The PA adversary  $B$  outputs a ciphertext  $c$  and the transcript  $\mathcal{L}^H, C$  of its interactions with the oracles.

Let  $K$  be some algorithm that outputs a decryption of  $c$  using the corresponding transcript  $\mathcal{L}^H, C$ . We restrict  $c \notin C$  to prevent trivially turning  $K$  into a decryption oracle. The plaintext awareness game is defined in figure 1.

---

### Algorithm 1 Plaintext awareness game

---

- 1:  $(\text{pk}, \text{sk}) \xleftarrow{\$} \text{KeyGen}(1^\lambda)$
  - 2:  $(\mathcal{L}^H, C, c) \leftarrow B^{H, \mathcal{E}_{\text{pk}}^H}(1^\lambda, \text{pk})$
  - 3:  $m \leftarrow K(1^\lambda, \text{pk}, \mathcal{L}^H, C, c)$
  - 4: **return**  $\llbracket c \notin C \wedge m = D(\text{sk}, c) \rrbracket$
- 

Figure 1: The plaintext awareness game

**Definition 1.1.** A public-key encryption scheme is plaintext-aware if there exists an efficient  $K$  such that for all efficient PA adversaries  $B$ , the probability of  $K$  failing to extract the correct decryption is negligible.

$$P[\text{PA}(K, B) \neq 1] \leq \text{negl}(\lambda)$$

Note that our definition of plaintext awareness deviates from [BDPR98] in that we do not require the PKE to also be IND-CPA secure. This is because in constructing a key encapsulation mechanism using the Fujisaki-Okamoto transformation, we only require the input PKE to be one-way secure, though under a stronger attack model. Restricting the definition to only the plaintext-awareness game allows us to combine PA with other security definition, such as OW-CPA, to prove stronger security result.

## 2 Encrypt-then-MAC transformations

Let  $\text{PKE}(\text{KeyGen}, \text{E}, \text{D})$  be a probabilistic public-key encryption scheme defined over message space  $\mathcal{M}_{\text{PKE}}$ , ciphertext space  $\mathcal{C}_{\text{PKE}}$ , and coin space  $\mathcal{R}_{\text{PKE}}$ . Where the encryption routine is deterministic, we simply set the coin space to contain a single element  $\mathcal{R} = \{r\}$ . Let  $\text{MAC}(\text{Sign}, \text{Verify})$  be a message authentication code defined over key space  $\mathcal{K}_{\text{MAC}}$ . The message space of the MAC should contain the ciphertext space of the PKE:  $\mathcal{C}_{\text{PKE}} \subseteq \mathcal{M}_{\text{MAC}}$ . Let  $G : \mathcal{M}_{\text{PKE}} \rightarrow \mathcal{R}_{\text{PKE}}$  and  $H : \mathcal{M}_{\text{PKE}} \rightarrow \mathcal{K}_{\text{MAC}}$  be hash functions.

The “encrypt-then-MAC” transformation  $\text{PKE}_{\text{EtM}}(\text{KeyGen}, \text{E}_{\text{EtM}}, \text{D}_{\text{EtM}}) = T_{\text{EtM}}(\text{PKE}, \text{MAC}, H)$  outputs a public-key encryption scheme where the key generation routine is identical to the input PKE's key generation routine. The de-randomized “encrypt-then-MAC” transformation  $\text{PKE}_{\text{EtM}}^{\$}(\text{KeyGen}, \text{E}_{\text{EtM}}^{\$}, \text{D}_{\text{EtM}}^{\$}) = T_{\text{EtM}}^{\$}(\text{PKE}, \text{MAC}, G, H)$  similarly outputs a public-key encryption scheme. In both transformations, the key generation routine remains unchanged. The modified encryption and decryption routines are described in figure 2 and 3.

---

**Algorithm 2**  $E_{\text{EtM}}(\text{pk}, m)$ 

---

```
1:  $r \xleftarrow{\$} \mathcal{R}_{\text{PKE}}$       $\triangleright$  If  $E$  is randomized, then  $E_{\text{EtM}}$  is randomized
2:  $k_{\text{MAC}} \leftarrow H(m)$ 
3:  $c \leftarrow E(\text{pk}, m; r)$ 
4:  $t \leftarrow \text{Sign}(k_{\text{MAC}}, c)$ 
5: return  $(c, t)$ 
```

---

---

**Algorithm 3**  $D_{\text{EtM}}(\text{sk}, (c, t))$ 

---

```
1:  $\hat{m} \leftarrow D(\text{sk}, c)$ 
2:  $\hat{k}_{\text{MAC}} \leftarrow G(\hat{m})$ 
3: if  $\text{Verify}(\hat{k}_{\text{MAC}}, c, t) \neq 1$  then
4:     return  $\perp$ 
5: end if
6: return  $\hat{m}$ 
```

---

Figure 2: “encrypt-then-MAC” transformation

---

**Algorithm 4**  $E_{\text{EtM}}^{\$}(\text{pk}, m)$ 

---

```
1:  $k_{\text{MAC}} \leftarrow H(m)$ 
2:  $r \leftarrow G(m)$ 
3:  $c \leftarrow E(\text{pk}, m; r)$ 
4:  $t \leftarrow \text{Sign}(k_{\text{MAC}}, c)$ 
5: return  $(c, t)$ 
```

---

---

**Algorithm 5**  $D_{\text{EtM}}^{\$}(\text{sk}, (c, t))$ 

---

```
1:  $\hat{m} \leftarrow D(\text{sk}, c)$ 
2:  $\hat{k}_{\text{MAC}} \leftarrow G(\hat{m})$ 
3: if  $\text{Verify}(\hat{k}_{\text{MAC}}, c, t) \neq 1$  then
4:     return  $\perp$ 
5: end if
6: return  $\hat{m}$ 
```

---

Figure 3: de-randomized “encrypt-then-MAC” transformation

**Theorem 2.1.** *If the MAC is one-time unforgeable and PKE is OW-CPA secure, then EtM and  $\text{EtM}^{\$}$  are both OW-CCA secure*

We will prove theorem 2.1 in two steps. First we show that EtM and  $\text{EtM}^{\$}$  are both plaintext-aware in section 2.1, then we show that plaintext awareness and OW-CPA security imply OW-CCA security in section 2.2.

The KEM transformations from [HHK17] require the input PKE to be OW-PCVA secure, although OW-PCVA security is implied by OW-CCA security

**Theorem 2.2.** *For every OW-PCVA adversary  $\mathcal{A}_{\text{PCVA}}$  against some PKE, there exists an OW-CCA adversary  $\mathcal{A}_{\text{CCA}}$  with identical advantage:*

$$\epsilon_{\text{OW-PCVA}} = \epsilon_{\text{OW-CCA}}$$

*Proof.* The OW-CCA adversary can perfectly simulate both the PCO and the CVO by replacing “running decryption routine” with “querying the decryption oracle”. When the OW-PCVA adversary halts, the OW-CCA adversary passes OW-PCVA adversary’s output as its own. Thus, OW-CCA adversary wins if and only if OW-PCVA adversary wins.  $\square$

## 2.1 EtM is plaintext aware

## 2.2 Plaintext awareness and OW-CPA imply OW-CCA

## References

- [BDPR98] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations among notions of security for public-key encryption schemes. In *Advances in Cryptology—CRYPTO’98: 18th Annual International Cryptology Conference Santa Barbara, California, USA August 23–27, 1998 Proceedings 18*, pages 26–45. Springer, 1998.
- [BR95] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In *Advances in Cryptology—EUROCRYPT’94: Workshop on the Theory and Application of Cryptographic Techniques Perugia, Italy, May 9–12, 1994 Proceedings 13*, pages 92–111. Springer, 1995.

- [HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the fujisaki-okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer, 2017.