Public-Key Encryption Resilient to Linear Related-Key Attacks

Hui Cui, Yi Mu, and Man Ho Au

School of Computer Science and Software Engineering, University of Wollongong, Wollongong, NSW 2522, Australia
hc892@uowmail.edu.au, \{ymu, aau\}@uow.edu.au

Abstract. In this paper, we consider the security of public-key encryption schemes under linear related-key attacks, where an adversary is allowed to tamper the private key stored in a hardware device, and subsequently observe the outcome of a public-key encryption system under this modified private key. Following the existing work done in recent years, we define the security model for related-key attack (RKA) secure public-key encryption schemes as chosen-ciphertext and related-key attack (CC-RKA) security, in which we allow an adversary to issue queries to the decryption oracle on the linear shifts of the private keys. On the basis of the adaptive trapdoor relations via the one-time signature schemes, Wee (PKC’12) proposed a generic construction of public-key encryption schemes in the setting of related-key attacks, and some instantiations from Factoring, BDDH with CC-RKA security, and DDH but with a weaker CC-RKA security. These schemes are efficient, but one-time signatures still have their price such that in some cases they are not very efficient compared to those without one-time signatures. Bellare, Paterson and Thomson (ASIACRYPT’12) put forward a generic method to build RKA secure public-key encryption schemes, which is transformed from the identity-based encryption schemes. However, so far, the efficient identity-based encryption schemes are generally based on pairings. To generate a specific construction of public-key encryption schemes against related-key attacks without pairings, after analyzing the related-key attack on the Cramer-Shoup basic public-key encryption scheme, we present an efficient public-key encryption scheme resilient against related-key attacks without using one-time signature schemes from DDH. Finally, we prove the CC-RKA security of our scheme without random oracles.

Keywords: Public-key encryption, Related-key attack, CC-RKA security.

1 Introduction

In the traditional security model, it is assumed that the adversary is isolated from the internal states of the honest communication parties. However, with the development of information technologies, the security of cryptographic algorithms in modern cryptography is analyzed in the black-box model, where an adversary may view the algorithm’s inputs and outputs, but the private key as
well as all the internal computation remains perfectly hidden. Unfortunately, this idealized assumption is often hard to satisfy in real systems. In many situations, the adversary might get some partial information about private keys through methods which were not anticipated by the designer of the system and, correspondingly, not taken into account when arguing its security. Such attacks, referred to as key-leakage attacks, come in a large variety. An important example is side-channel [18] attacks that exploit information leakage from the implementation of an algorithm, where an adversary observes some “physical output” of a computation (such as radiation, power, temperature, running time), in addition to the “logical output” of the computation.

In recent two decades, this requirement has been relaxed to capture security under the scenarios where some information of the keys is leaked to the adversary. When an adversary tampers the private key stored in a cryptographic hardware device, and observes the result of the cryptographic primitive under this modified private key, there is a related-key attack (RKA) [4,11]. The key here could be a signing key of a certificate authority or a decryption key of an encryption scheme. In related-key attacks, the adversary attempts to break an encryption scheme by invoking it with several private keys satisfying some known relations.

Wee [20] proposed a generic construction of public-key encryption schemes in the setting of linear related-key attacks. In [20], the constructions exploit certain existing public-key encryption schemes that are susceptible to linear related-key attacks, to obtain public-key encryption schemes that are secure against linear related-key attacks from adaptive trapdoor relations via strong one-time signatures, which generates a tag in the ciphertext of the concrete scheme. The security of this realization is analogous to those for obtaining chosen-ciphertext attack (CCA) security from extractable hash proofs [19], and trapdoor functions [15], which implies a trick that the RKA decryption oracle will return ⊥ for tag = tag* generated from an one-time signature scheme, whenever the ciphertext with tag given by the adversary matches the challenge ciphertext with tag* or not. Briefly, RKA.Decrypt oracle outputs ⊥ when given a ciphertext with tag = tag* even φ(sk) ≠ sk, where φ denotes a linear shift. That is to say, the RKA decryption query will not help the adversary to obtain more information if tag = tag*. Besides, Wee [20] designed some efficient strong one-time signatures to reduce the total overhead of the specific schemes. However, though one-time signatures are easy to construct in theory, and are more efficient than full-fledged signatures, they still have their price. Particularly,

- Known one-time signature schemes based on general one-way functions [10] allow very efficient signing, key generation and signature verification, but they require the expensive valuations of the one-way function. More problematic, such schemes usually have long public keys and signatures, resulting in long ciphertexts.
- Although one-time signature schemes constructed based on number-theoretic assumptions by adapting full-fledged signature schemes have the advantage of shorter public keys and signatures, but this yields schemes of which computational cost for key generation, signing, and verifying is more expensive.