Non-Malleability: An Introduction and Survey of Recent Developments

Cynthia Dwork* Moni Naor†

1 Instead of an Introduction: Non-Malleability by Example

In their seminal and beautiful paper *Probabilistic Encryption*, Goldwasser and Micali laid the theoretical groundwork for modern cryptography, both giving a clean definition of secrecy in a complexity-theoretic framework, and providing a candidate public-key cryptosystem generator that, under prevailing computational assumptions, satisfies that definition. For the case of a passive eavesdropper listening to an encrypted conversation, their notion, called *semantic security*, is still the "gold standard" of security. One simple and intuitive description of semantic security is

> For all relations R, seeing $\alpha \in E(x)$ does not help one to find y such that $R(x, y)$ holds[1].

In other words, the adversary "learns nothing" about the plaintext (it is assumed that the adversary is restricted to probabilistic polynomial time computations).

More than 20 years later, cryptography is now routinely used as a tool among *participants* in a protocol, not just to hide information from outside eavesdroppers but also from malicious participants in the protocol. In this case, the adversary is no longer passive, but active – and semantic security no longer suffices.

To see this, consider the following toy coin-flipping protocol. There are three players: Alice, Bob, and a judge. The goal is for Alice and Bob to produce an unbiased bit with the help of the judge (who for some reason cannot just flip a coin and announce the result). The judge chooses a (public key, private key) pair using a public-key cryptosystem generator, and publishes E, the encryption algorithm. First Alice, and then Bob, chooses a random bit, encrypts it using E, and sends the result to the judge. The judge decrypts and announces the two bits, and the result is the exclusive-OR of these bits.

The (erroneous) intuition is that, since Bob does not know Alice's bit, any ciphertext he produces is the encryption of a bit that is *independent* of Alice's bit, and so the exclusive-OR is unbiased. But suppose Bob could ensure that his bit is the same as Alice's; then the outcome will always be 0, *even if Bob has no idea what he is saying*. Similarly, if Bob could ensure that his bit is always the opposite of Alice's, he can force an outcome of 1. It is a straightforward exercise for Bob to achieve either of these goals if the cryptosystem used is the one proposed in [28], despite the semantic security of the scheme.

As the above example shows, something stronger than semantic security is required; we call this property *non-malleability*:

> For all polynomial time computable relations R, seeing $\alpha \in E(x)$ does not help one to find $\beta \in E(y)$, where $\beta \neq \alpha$, such that $R(x, y)$ holds.

*Microsoft Research, Silicon Valley Campus, 1065 L'Avenida, Mountain View, CA 94043. Email: dwork@microsoft.com.

†Incumbent of the Judith Kleeman Professorial Chair, Department of Computer Science and Applied Math, Weizmann Institute of Science, Rehovot 76100 Israel. Email: naor@wisdom.weizmann.ac.il. Research supported in part by a grant from the Israel Science Foundation.

[1]By $\alpha \in E(x)$ we mean that α is an element of the set of legal encryptions of x.

In other words, a polynomial time bounded adversary cannot generate an *encryption* of a plaintext value related to x.

Just as semantic security specifies one kind of break of a cryptosystem (a break is a violation of semantic security), non-malleability, or its opposite, malleability, specifies an alternate notion of what it means to break a cryptosystem.

In addition to specifying what it means to break a cryptosystem, one must also specify what are the means of attack available to the adversary. We consider non-malleability of public-key cryptosystems under three kinds of attack already defined in the literature: chosen plaintext attacks[2], chosen ciphertext attacks in the preprocessing mode, and chosen ciphertext attacks in the postprocessing mode. In preprocessing mode, denoted *cca-pre*, the adversary may make polynomially many queries of a decryption oracle before being presented with a challenge ciphertext (which it then tries to *maul*, *i.e.*, for which it tries to find a violation of non-malleability). In the postprocessing mode, denoted *cca-post*, the adversary carries out a preprocessing attack, is presented with a challenge ciphertext α, and is then permitted the postprocessing attack: it may query the decryption oracle with any polynomial number of ciphertexts *other than the exact string* α before trying to maul the ciphertext.

Given two types of requirements (semantic security and non-malleability) and three types of attacks (chosen plaintext, cca-pre, cca-post), Remark 3.6 and [2] gives a complete characterization of the relationships between the six possible combinations. Interestingly, under the strongest attack, chosen ciphertext in the postprocessing mode, non-malleability and semantic security are equivalent. This is not the case for weaker attack models.

The concept of non-malleability has application beyond encryption. For instance, in the toy coin-flipping protocol described above, one role of the encryption step is to *commit* to the bits before opening them and computing their exclusive-OR. In the toy protocol, the opening of the commitment is made trivial by the presence of the judge; in a less centralized protocol the commitment would be done by secret sharing. But the problem of malleability is the same: *secrecy* of Alice's committed value does not ensure *independence* from the value committed to by Bob. If Bob can maul Alice's commitment messages so as to commit to the same or opposite value at will, the fact that Alice's value is secret is irrelevant.

A natural response is to require Alice and Bob to prove *knowledge* of their committed values: intuitively, if Bob *knows* what he is saying (this can be made precise), then his bit must be independent of Alice's bit – otherwise we would have a violation of semantic security. But again, there is a malleability problem: perhaps the proof of knowledge itself is malleable, even if it is a zero-knowledge proof of knowledge.

Accordingly, in addition to non-malleable public-key encryption under all three forms of attack, as well as non-malleability of shared-key encryption, this paper defines and constructs non-malleable commitment and non-malleable zero-knowledge interactive proofs. All our constructions are obtained under general assumptions.

2 What's New on Our Three Problems?

We begin with developments in non-malleable encryption, specifically, in public-key cryptosystems that are non-malleable under chosen ciphertext attack in the post-processing mode (nm-cca-post).

Let us review the Naor-Yung cryptosystems that are semantically secure under chosen ciphertext attacks in the *preprocessing* mode [35]. Naor and Yung begin with any public-key cryptosystem that is semantically secure under chosen *plaintext* attack. Let $\mathcal{G}$ be the cryptosystem generator for this underlying system. A public key in the Naor-Yung construction consists of a pair of public keys obtained from the generator $\mathcal{G}$,

[2]This is the weakest attack that makes sense against a public-key cryptosystem, since an eavesdropper can generate encryptions of plaintexts of her choice using the public encryption key.

together with a reference string σ for a noninteractive zero-knowledge proof system (NIZK). The encryption process is

- Compute $\alpha_1 \in_R E_1(m)$, $\alpha_2 \in_R E_2(m)$;
- Construct a NIZK, using σ, proving that α_1 and α_2 encrypt the same plaintext; we call this a proof of *consistency*.

Thus, the approach can be summarized as "add redundancy and prove consistency[3]."

The Naor-Yung scheme is malleable if the encryptions and the NIZK are malleable. We addressed this difficulty by arranging that the ciphertext incorporate a sort of *untamperable authentication* as follows:

1. The public key consists of n ***pairs*** of public keys, $(E_0^1, E_1^1), \ldots, (E_0^n, E_1^n)$, drawn according to $\mathcal{G}$, together with a reference string σ for NIZKs.
2. - Choose an instance of a *digital signature* scheme;
   - View the public verification key of the signature scheme as sequence of bits selecting, for each pair of keys in the public key, one key under which to encrypt the plaintext;
   - Encrypt the plaintext m under each of the selected keys;
   - Provide a NIZK of consistency of encryptions using the reference string σ.
3. The ciphertext consists of the public verification key for the signature scheme, the n ciphertexts, and the proof of consistency of encryption.

The principal innovation is in the use of the freshly-chosen (that is, chosen anew *by the sender* for each message to be encrypted) signature scheme to authenticate the ciphertext. The intuition for non-malleability is straightforward. Assume the attacker is given a ciphertext α that it wishes to maul. If the attacker uses the same instance of the signature scheme as was used in generating α, then it will not know the secret signing key, and so will be unable to generate a signature on any content other than the content already signed in α, preventing it from generating a different valid ciphertext. If, instead, the attacker changes the signature scheme, then there will be at least one pair of encryption keys so that, without loss of generality, α contains an encryption $E_0^i(m)$, and the adversary must generate $E_1^i(m')$ for some m' related to m (possibly $m' = m$); intuitively, since the two keys are chosen completely independently, the attacker has been given no clue how to generate an encryption of a related m' under E_1^i (even if it could easily do so under E_0^i because it has seen $E_0^i(m)$ and encryptions under any individual key may be mauled).

In fact, to our knowledge, all general constructions of nm-cca-post cryptosystems, share the following paradigm: The attacker sees a ciphertext and wishes to generate a ciphertext of a (polynomially related) message. The goal is to make this difficult, even given access to a decryption oracle.

1. NIZK proofs are used to prove that the ciphertext is valid. Thus no information is gained from a "garbage / not garbage" oracle (this is also true of the ss-cca-pre scheme in [35]);
2. Some portion of the ciphertext is authenticated with a signature under a freshly chosen instance of a (one-time) signature scheme. The intuition is that the signed content and the choice of public signature verification key are inextricably linked: changing the verification key changes the content to be signed so that an attacker will be unable to generate the necessary content; leaving the verification key unchanged ensures that the attacker will not have the necessary private signing key to sign anything new.

[3]Decryption proceeds as follows: *first* check the proof of consistency and refuse to answer if the proof cannot be verified; otherwise decrypt either ciphertext.

In our construction, the public verification key of the signature scheme selects among a set of encryption keys under which consistent encryptions must be generated, together with a NIZK proof of consistency. In 1999 Sahai presented an alternate scheme in which the public encryption key consists of only two encryption keys, but has multiple reference strings for NIZKs [42]. In his case, the public verification key for the signature scheme is used to select among the set of reference strings; a single pair of encryptions are proved consistent using NIZKs under all of the selected reference strings. Thus, in terms of the Naor-Yung approach ("add redundancy and prove consistency"), we directly made encryption *per se* non-malleable, while Sahai made the proof of consistency non-malleable.

Sahai's work drew attention to a technical aspect in the definition of NIZKs arising in proving his construction correct. Exploration of this technicality resulted in several new constructions, all based on general assumptions (see in particular [16], who obtain NIZK arguments that remain non-malleable with regard to any polynomial number of proofs and, extending the results of [15], bounded NIZK *arguments of knowledge*). Perhaps the conceptually simplest of these is due to Lindell [33]; his writeup is exceptionally clear.

2.1 Validity Checks: Power of the Garbage / Not Garbage Oracle

When an attacker carries out a chosen-ciphertext attack, the purported ciphertexts presented to the decryption oracle need not be syntactically well-formed. In this case, the attacker may obtain an "invalid" response from the oracle. As first noted by Goldwasser, Micali, and Tong [29], this is very valuable information. Indeed, Bleichenbacher [5] was able to use this information to decrypt arbitrary ciphertexts in a cca-post attack on PKCS # 1 standard, and Boyarsky [7] showed how to exploit this information to break, in a multi-user environment, the password authentication mechanism proposed by [30].

Accordingly, all the chosen ciphertext resilient schemes discussed so far have the property that anyone, including a passive observer who has no idea how a purported ciphertext string was created, can tell if the ciphertext is well-formed (all the schemes use NIZKs of consistency, which can be verified by anyone). This means no information can be leaked if the decryption oracle says "invalid." A different approach to this problem was taken by Cramer and Shoup, who created the first *efficient* nm-cca-post cryptosystem [12] (it is efficient in the sense that it is only twice as expensive (computationally) than the best ss-CPA cryptosystem). In this scheme, which is based on the Decisional Diffie Hellman assumption (see also [13] for Quadratic Residuousity and Pallier based versions, and a generalization of the basic technique), messages are accompanied by *tags*. Unlike the case of NIZKs of consistency, which can be verified by anyone, the verification of the tags by the decryption oracle is based on a *secret* function applied to the ciphertext. That is, the legitimate receiver computes a secret function of the ciphertext and compares the result to the tag. If the two differ, then the ciphertext is rejected as invalid (hence no public verification of the validity of the ciphertext is possible). The key here is that, roughly speaking, an attacker does not know the secret information and hence cannot create invalid ciphertexts with tags that pass the test. Thus, if the adversary starts with some ciphertext and modifies it, the resulting ciphertext is almost surely invalid and no information is gained by learning this fact from the decryption oracle. In a little more detail (see [13]), the ciphertext encrypting a message m is generated using a pair (x, w) obtained using a public sampling algorithm, where w is a witness to $x \in L$ for a certain language L. Without w it is computationally hard to generate a valid ciphertext; with w it is easy to deduce m from the ciphertext.

Canetti and Goldwasser constructed a *threshold* version of the original Cramer-Shoup cryptosystem [11]. One of the problems they needed to overcome was the fact that in the Cramer-Shoup system there is no public verification of the validity of a ciphertext. A key step in their construction was to modify the response to an invalid message by returning a *random* plaintext (instead of the text "invalid"). This modification is easily shown not to weaken the original system. On the other hand, it greatly simplifies distributed treatment of

the validity check.

2.2 Non-Malleable Commitments

In 2002 Barak reduced the number of rounds needed for non-malleable commitment from logarithmic in the security parameter to constant [1]. His approach was to reduce the problem to that of achieving non-malleable string commitment in the *shared random string* model, in which others had already obtained non-malleable commitment [18] (see also [17] for a construction satisfying a weaker notion of non-malleability in this model) and non-malleable NIZKs. The key, then, was to devise an appropriate coin-flipping protocol that achieves the effect of a shared random string without assuming the existence of a trusted party to establish the string. More specifically, consider the scenario in which there are four parties A, B, C, and D. A is committing to a value in conversation with B, while C is committing to a value in conversation with D. Assuming B and C are controlled by an adversary, the goal is to force the value committed to by C to be either identical to the value committed to by A or independent of it. Barak turns this into a coin-flipping problem in which each of the pairs (A, B) and (C, D) first (but asynchronously) chooses a shared random string, and the strings are then used for commitments. Non-malleability is achieved whenever the adversary is effectively constrained to choose only among the following two possibilities: (1) the two strings are identical and (2) the two strings are uniformly and independently chosen.

3 What's New in Related Problems

3.1 Variants of Non-Malleable Commitment

Our approach to non-malleability can be characterized as "think globally, act locally." In related work, others have assumed the existence of a shared random string, or other public parameters, an approach that might be characterized as "think globally, act globally." We discuss some of these results here.

Di Crescenzo, Ishai, and Ostrovsky were the first to examine non-malleable commitment in the shared random string model. In addition, they weakened the non-malleable commitment problem as follws. While we require, roughly, that the adversary not be able to successfully *commit* to a value related to another committed value, they require that the adversary not be able to *open* its commitment to a value related to another committed value [17]. In this setting, they achieve non-interactive non-malleable commitments with respect to opening, based on any one-way function.

Fischlin and Fischlin were the first to write about the fact that the non-malleability condition satisfied in [17] was actually different than the condition satisfied in our original work [23][4]. Observing that non-malleability with respect to opening is the only notion that makes sense in an information-theoretically secret commitment, they constructed efficient but interactive protocols in the so-called *public parameter* model. In this model certain parameters (not just a random string but, more typically, a public encryption key or some primes and generators of a group) are assumed to have been selected and made public by some *deus ex machina*. Their schemes rely on the RSA and discrete logarithm assumptions, respectively.

Continuing research on non-malleability with respect to opening in the public parameter model, Di Crescenzo, Katz, Ostrovsky [18], and Smith obtained very efficient non-interactive solutions, based on the same assumptions. (See also the work of Damgard and Groth for similar results under more general assumptions (existence of one-way functions), and *reusability* of the shared random string [14].) They also proved correct a charmingly simple construction for the *perfectly binding* case[5], based on any public-key

[4] Although the weakened version is frequently sufficient, there are cases in which this is not true [25].

[5] In other words, commitment is absolute; there only exists one way in which the commitment can be opened, independent of the computational power of the committer.

cryptosystem that is non-malleable under chosen *plaintext* attacks (easier to achieve than nm-cca-post), combined with any shared-key cryptosystem that enjoys indistinguishability under plaintext oracle cca-post attacks (here, the adversary is given access to a *de*cryption oracle, but not to an *en*cryption oracle). In the perfectly binding commitment problem, the length of the (conversation yielding) commitment cannot be shorter than the length of the value to which one is committing; this is no longer the case with perfectly hiding commitments. The length of the commitments in [18] is $3k$, where k is the security parameter, independent of the (polynomial in k) length of the messages, and the length of the public parameters is $O(k)$. The scheme can be based either on the RSA or discrete logarithm assumptions.

As noted in [18], both their perfectly hiding schemes and the commitment scheme in [17] follow the following paradigm (all these schemes are non-interactive):

> "A commitment consists of three components $< A, B, Tag >$. The first component A is a commitment to parameters r_1 and r_2 for a one-time "message authentication code" (MAC) for B. The second component B contains the actual commitment to the message m, using public parameters which depend upon the first component A. Finally, $Tag = MAC_{r_1,r_2}(B)$. An adversary who wishes to generate a commitment to a related value has two choices: he can either re-use A or use a different A'. If he re-uses A, with high probability he will be unable to generate a correct *Tag* for a different B', since he does not know the values r_1, r_2. On the other hand, if he uses a different A', the public parameters he is forced to use for his commitment B' will be different from those used for the original commitment."

Compare this to the paradigm we sketched for general constructions of nm-cca-post public-key cryptosystems: By viewing A as a commitment to a signing key (A can even be the corresponding public verification key), and viewing B as (a portion of) the ciphertext, we have that the *Tag* corresponds to a signature on the ciphertext under the committed signing key.

3.2 Random Oracles and Efficiency

If we make the (drastically) simplifying assumption that a specific function behaves as an idealized random function (random oracle), then it possible to obtain simple and efficient constructions of public-key encryption schemes that are secure against cca-post attacks [4, 2, 37, 26, 6, 24]. The meaning of such results is the subject of much debate (see [10, 21, 27, 3].) However, it is not known whether a generic and efficient method exists, under the random oracle assumption, for converting every public-key cryptosystem that semantically secure (or that satisfies even some weaker property) against chosen plaintext attacks into one that is secure against cca-post attacks. It is conceivable that such a method exists. Among the problems in applying approaches such as OAEP and its variants, REACT, and Fujisaki-Okamoto, are that in the underlying "input" cryptosystem (1) there can exist ciphertexts which can be validly created from two different plaintext message; (2) the decryption mechanism may not return "invalid" on all invalid ciphertexts. Proos [38] and Jaulmes and Joux [31] exploited these issues in attacking variants of NTRU.

3.3 Interactive Encryption:

Katz [32] defined non-malleable proofs of knowledge and showed how these can be efficiently constructed based on the RSA assumption. Three applications described in [32] are *interactive* public-key cryptosystems, password authentication, and deniable authentication, which we discuss next. As Katz notes, in a protocol that requires interaction anyway, it may perfectly acceptable to replace a non-interactive encryption scheme with an interactive one.

4 An Application: Deniable Authentication

An authentication protocol allows a receiver of a message, Bob, to verify that the message received was indeed sent by the claimed sender, Alice. Authentication is one of the fundamental applications of cryptography, and is trivially achievable given digital signatures; indeed, one of the key insights in the seminal paper of Diffie and Hellman [19] was the idea that authentication can be made *transferable*, *i.e.*, Bob not only knows the message came from Alice but he can convince a third party that this is indeed the case. The transferability property requires that Alice have a public (verification) key, as well as a secret (signing) key allowing her to produce a digital signature of the message.

But is transferability of authentication always desirable? One need not be a card carrying member of the EFF[6] to appreciate that not everything we ever say should be transferable to anyone else – but this is precisely the case as more and more of our interactions move on-line. With this issue in mind, we gave the following simple construction of a public-key authentication mechanism from any non-malleable cca-post cryptosystem: Alice publishes a public encryption key E for an nm-cca-post cryptosystem. When Bob wants Alice to authenticate a message m, he chooses a random suffix r and sends to Alice $c \in_R E(m \circ r)$ (here "$\circ$" denotes concatenation). Alice responds with r if and only if she is willing to authenticate m.

In Non-Malleable Cryptography [20] we did not formally define deniable authentication, and our scheme was in fact only provably deniable against "honest" receivers. Intuitively, deniability is easily defined and achieved via zero-knowledge: if the interaction is zero-knowledge, then it is surely deniable, since Bob could create simulated "transcripts" of conversations with Alice for messages of his choice. Pursuit of this idea lead to the work on concurrent zero-knowledge ([22]) described in Section 5.

A related question is whether it is possible to hide the authenticator's identity *even from the intended recipient*. For example, Alice, as a member of the cabinet, may wish to leak certain economic proposals to the press (Bob), revealing only her status as a cabinet member, and not her precise identity. This is a serious application, something we view as an important part of the checks and balances that monitor an open society. Keeping the sender's identity secret while being sure that it is a valid confirmation of the message may sound paradoxical, since the receiver verifies the authenticity of the message with respect to some public information related to the party doing the authentication (*e.g.*, a public key). However a method for doing just that was suggested by Rivest, Shamir and Tauman [40]. They proposed the notion of *Ring Signatures* (a generalization of group signatures of Chaum and van Heyst [8])allowing a member of an *ad hoc* collection of users S (such as high ranking government officials), to prove that a message is authenticated by a member of S. The assumption is that each member of S has published a public signature key of a scheme with certain properties (where RSA and Rabin are examples). The construction given in [40] is very efficient, but its analysis is based on the ideal cipher model (a strengthening of the random oracle one.)

Naor [34] proposed a notion that merges Ring Signatures and Deniable Authentication to form *Deniable Ring Authentication*. Roughly speaking, a deniable ring authentication scheme enables the sender, for any message she wishes and for any ad hoc collection S of users containing the sender, to deniably (in the zero-knowledge sense) prove that a member of S is the one authenticating the message. Moreover, the authentication leaks absolutely no information about *which* member of S is doing the authenticating (any two members of S generate conversations that are indistinguishable to the recipient). The verifier need not be "part of the system" or have public keys of his own.

[6] Electronic Frontier Foundation.

5 Plug and Play?

Our work on non-malleability was an early step toward the general goal of constructing cryptographic primitives that interoperate in an intuitive and modular fashion. To some extent non-malleability focusses on protecting the *receiver* – of an encrypted message, a committed value, or a zero-knowledge proof. *Concurrent* zero-knowledge, introduced by Dwork, Naor, and Sahai, shifts attention to the *sender*, or, more precisely, a prover simultaneously engaged in many executions of a zero-knowledge proof [22]. The problem is that when several executions of zero-knowledge protocols are running concurrently, the execution may not be zero-knowledge *in toto*. Dwork *et al.* proposed explicit use of certain local timing constraints in interactive protocols. The first concurrent zero-knowledge interactive arguments for all of NP, without timing constraints, were constructed by Richardson and Kilian [39]. This has been an intense area of research, and a survey is beyond the scope of these remarks (see Rosen [41]).

Canetti initiated study of a different paradigm, *universal composability* [9]. This is a general methodology for expressing security requirements; Canetti showed that for protocols satisfying these requirements, security is preserved under arbitrary composition. Canetti's positive results in this direction are in the shared random string model, and this is essential for most interesting tasks in his framework. Universal composability is increasingly widely studied, and cannot be surveyed here.

6 Conclusions

The issue of malleability is really one of inter-operability prevalent throughout the various tasks of cryptography. We believe that inter-operability of cryptographic primitives is the most pressing practical challenge to cryptography today, and that the theoretical groundwork for facing this challenge is far from complete.

References

[1] B. Barak, Constant-Round Coin-Tossing with a Man in the Middle or Realizing the Shared Random String Model, *Proc. IEEE FOCS 2002*, pp. 345–355

[2] M. Bellare, A. Desai, D. Pointcheval and P. Rogaway, Relations Among Notions of Security for Public-Key Encryption Schemes, Advances in Cryptology - CRYPTO '98 (1998), vol. 1462 of LNCS, Springer-Verlag, pp. 2646.

[3] M. Bellare, A. Boldyreva and A. Palacio A Separation between the Random-Oracle Model and the Standard Model for a Hybrid Encryption Problem, Cryptology ePrint Archive.

[4] M. Bellare and P. Rogaway, P. Optimal Asymmetric Encryption.In Advances in Cryptology - EUROCRYPT '94 (1995), vol. 950 of LNCS, Springer-Verlag, pp. 92111.

[5] D. Bleichenbacher, Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1, Advances in Cryptology – CRYPTO '98, Lecture Notes in Computer Science, pp. 1–12.

[6] Dan Boneh, *Simplified OAEP for the RSA and Rabin Functions*, Advances in Cryptology - CRYPTO 2001, LNCS2139, Springer 2001, pp. 275–291.

[7] M. K. Boyarsky, Public-Key Cryptography and Password Protocols: The Multi-User Case, *Proc. ACM Conference on Computer and Communications Security 1999* pp. 63–72

[8] D. Chaum and E. van Heyst, *Group Signatures*, Advances in Cryptology - EUROCRYPT'91, LNCS 541, Springer, 1991, pp. 257–265.

[9] R. Canetti, Universally Composable Security: A New Paradigm for Cryptographic Protocols, *Proc. IEEE FOCS 2001*, pp. 136–145

[10] R. Canetti, O. Goldreich, and S. Halevi, The random oracle methodology, revisited, Proceedings of the 30th Annual ACM Symposium on the Theory of Computing, 1998, pp. 209–218.

[11] R. Canetti and S. Goldwasser, An Efficient *Threshold* Public Key Cryptosystem Secure Against Adaptive Chosen Ciphertext Attack, Advances in Cryptology – Eurocrypt '99, Lecture Notes in Computer Science, pp. 90–106, 1999

[12] R. Cramer and V. Shoup, A Practical Public Key Cryptosystem Secure Against Adaptive Chosen Ciphertext Attacks, Advances in Cryptology – CRYPTO '98, Lecture Notes in Computer Science, vol. 1462, Springer, 1998, pp. 13–25.

[13] R. Cramer and V. Shoup, Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption, Advances in Cryptology – EUROCRYPT 2002, LNCS 2332, pp. 45–64, Springer Verlag, 2002.

[14] I. Damgard and J. Groth, Non-Interactive and Reusable Non-Malleable Commitment Schemes, Proceedigns of the 35th Annual ACM Symposium on Theory of Computing, 2003, pp. 426–437.

[15] A. DeSantis and G. Persiano, Zero-Knowledge Proofs of Knowledge Without Interaction, *Proceedings IEEE FOCS 1992*

[16] A. De Santis, G. Di Crescenzo, R. Ostrovsky, G. Persiano, A. Sahai, Robust Non-interactive Zero-Knowledge, Advances in Cryptology – CRYPTO 2001, Springer, 2001, pp. 566–598.

[17] G. Di Crescenzo, Y. Ishai, R. Ostrovsky, Non-Interactive and Non-Malleable Commitment, Proceedings of the 30th Annual ACM Symposium on the Theory of Computing 1998, pp. 141–150.

[18] G. Di Crescenzo, J. Katz, R. Ostrovsky, and A. Smith, Efficient and Non-interactive Non-malleable Commitment, *Proc. EUROCRYPT 2001*, pp. 40–59

[19] W. Diffie, and M.E. Hellman. New Directions in Cryptography. *IEEE Trans. on Info. Theory*, IT-22 (Nov. 1976), pages 644–654.

[20] D. Dolev, C. Dwork and M. Naor, *Non-malleable Cryptography*, Siam J. on Computing, vol 30, 2000, pp. 391–437.

[21] C. Dwork, M. Naor, O. Reingold, L. J. Stockmeyer, Magic Functions, Proc. IEEE FOCS 1999, pp. 523–534.

[22] C. Dwork, M. Naor, and A. Sahai, Concurrent Zero-Knowledge, Proc. Proceedings of the 30th Annual ACM Symposium on the Theory of Computing, 1998, pp. 409–418

[23] M. Fischlin and R. Fischlin, Efficient Non-Malleable Commitment Schemes, Advances in Cryptology – CRYPTO 2000, LNCS vol. 1880, Springer, 2000, pp. 413–431.

[24] E. Fujisaki and T. Okamoto, How to Enhance the Security of Public-Key Encryption at Minimum Cost. In PKC '99 (1999), vol. 1560 of LNCS, Springer-Verlag, pp. 5368.

[25] Observation attributed to O. Goldreich and Y. Lindell in [23].

[26] E. Fujisaki, T. Okamoto, D. Pointcheval, J. Stern, *RSA-OAEP Is Secure under the RSA Assumption* Advances in Cryptology – CRYPTO 2001, Springer, 2001, pp. 260–274.

[27] Shafi Goldwasser and Yael Tauman, On the (In)security of the Fiat-Shamir Paradigm, Cryptology ePrint Archive: Report 2003/034. To appear, FOCS 2003.

[28] S. Goldwasser and S. Micali, Probabilistic encryption, *J. Comput. Syst. Sci.* 28 (1984), pp. 270–299.

[29] S. Goldwasser, S. Micali, and P. Tong, Why and How to Establish a Private Code on a Public Network, *Proc. IEEE FOCS 1982*, pp. 134–144.

[30] S. Halevi and H. Krawczyk, Public-key Cryptography and Password Protocols, *ACM Transactions on Information and System Security 2*(3), 1999, pp. 230–268.

[31] E. Jaulmes and A. Joux, A Chosen Ciphertext Attack on NTRU, Advances in Cryptology – CRYPTO'2000, LNCS 1880, Springer, 2000, pp. 20–35.

[32] J. Katz, Efficient and Non-Malleable Proofs of Plaintext Knowledge and Applications, Advances in Cryptology – EUROCRYPT 2003, LNCS 2656, Springer, 2003, pp. 211–228 .

[33] Y. Lindell, A Simpler Construction of CCA2-Secure Public-Key Encryption Under General Assumptions, Advances in Cryptology—Proceedings Eurocrypt 2003, LNCS 2656, 2003, pp. 241–254.

[34] M. Naor, Deniable Ring Authentication, Advances in Cryptology –CRYPTO 2002, Springer, 2002, 481–498.

[35] M. Naor and M. Yung, Public-Key Cryptosystem Secure Against Chosen Cipher-Text Attacks, Proceedings 22nd Annual ACM Symposium on Theory of Computing STOC 1990, pp. 427–437.

[36] T. Okamoto and D. Pointcheval, REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform. In Proc. of CT-RSA'01 (2001), vol. 2020 of LNCS, Springer-Verlag, pp. 159175.

[37] V. Shoup, OAEP Reconsidered. Journal of Cryptology 15(4): 223-249 (2002).

[38] J. Proos, Imperfect Decryption and an Attack on the NTRU Ecnryption Scheme, IACR Cryptlogy Archive, Report 02/2003.

[39] R. Richardson and J. Kilian, On the Concurrent Composition of Zero-Knowledge Proofs, Advances in Cryptology - EUROCRYPT 1999, pp. 415–431.

[40] R. L. Rivest, A. Shamir, and Y. Tauman, *How to Leak A Secret*, Advances in Cryptology - ASIACRYPT 2001, Lecture Notes in Computer Science, Vol. 2248, Springer, pp. 552–565.

[41] A. Rosen, The Round-Complexity of Black-Box Concurrent Zero-Knowledge, PhD Dissretation, Weizmann Institute, 2003.

[42] A. Sahai, Non-Malleable Non-Interactive Zero Knowledge and Adaptive Chosen-Ciphertext Security, *Proc. IEEE FOCS 1999*, pp. 543–553