Adaptive Proofs have Straightline Extractors (in the Random Oracle Model)

David Bernhard, Ngoc Khanh Nguyen, and Bogdan Warinschi

Computer Science Department
University of Bristol

{csxdb,nn14160,csxbw}@bristol.ac.uk

Abstract. The concept of *adaptive security* for proofs of knowledge was recently studied by Bernhard et al. They formalised adaptive security in the ROM and showed that the non-interactive version of the Schnorr protocol obtained using the Fiat-Shamir transformation is not adaptively secure unless the one-more discrete logarithm problem is *easy*. Their only construction for adaptively secure protocols used the Fischlin transformation [11] which yields protocols with *straight-line extractors*. In this paper we provide two further key insights. Our main result shows that any adaptively secure protocol must have a straight-line extractor: even the most clever rewinding strategies cannot offer any benefits against adaptive provers. Then, we show that any Fiat-Shamir transformed Σ-protocol is not adaptively secure unless a related problem which we call the Σ-one-wayness problem is *easy*. This assumption concerns not just Schnorr but applies to a whole class of Σ-protocols including e.g. Chaum-Pedersen and representation proofs. We also prove that Σ-one-wayness is hard in an extension of the generic group model which, on its own is a contribution of independent interest. Taken together, these results suggest that the highly efficient proofs based on the popular Fiat-Shamir transformed Σ-protocols should be used with care in settings where adaptive security of such proofs is important.

1 Introduction

Noninteractive zero knowledge proofs are a useful tool widely deployed in modern cryptographic constructions. They allow a prover (e.g. the creator of a ciphertext) to send a single message which will convince a verifier of the veracity of a certain statement (e.g. that the plaintext underlying a ciphertext respects certain constraints), without revealing any further information. A particularly useful variant are the so called “proofs of knowledge" where there exists an extractor who can efficiently recover from the prover a witness that the statement is true. Over parties in the system (who should obtain no information from the witness), the extractor benefits from setup assumptions like the common reference string model or the random oracle mode [3]. Our focus is on the latter model which yields by far the most efficient noninteractive proofs of knowledge

known to date. In a nutshell, in this paper we study different extraction strategies afforded by the random oracle model and identify fundamental efficiency limitations that such proofs need to face in practically relevant scenarios.

Background. Recall that in the random oracle setting the extractor learns the RO answers/queries that the prover makes. Here, we distinguish between several extraction strategies. [1] An extractor is *straight-line* if it only sees a single execution of the prover. An extractor is *rewinding* if it is allowed to launch and interact with further copies of the prover (with the same coins used to produce the statement) before returning a witness.

The distinction between straightline and rewinding extractors may be crucial in applications since for a rewinding extractor it is not clear how many times does it have to rewind to extract all witnesses from a prover who makes a sequence of n proofs. Shoup and Gennaro [20] first encountered this problem in the context of proving CCA security of a particular public-key encryption scheme; the "obvious" approach ends up rewinding 2^n times which leads to an inefficient reduction. Clearly, this problem disappears for a straight-line extractor.

The notion of *adaptive* proofs which lie somewhere between proofs with inefficient rewinding strategies and straight-line PoKs has been recently proposed by Bernhard et al. [5]. A proof scheme is adaptively secure if there is an extractor that can rewind, but must efficiently extract even from provers who make sequences of proofs. The notion is called adaptive because the extractor must return a witness for the first proof to the prover before the prover makes the second one, and so on.

As an application, Bernhard et al. study the Fiat-Shamir-Schnorr proofs: a non-interactive proof of knowledge obtained by applying the popular Fiat-Shamir [10] transform to the Schnorr [17] protocol for proving knowledge of discrete logarithm. The main theorem of [5] shows the Fiat-Shamir-Schnorr proof scheme is *not* an adaptive proof unless the one-more discrete logarithm problem is easy. This result essentially separates the usual PoK notion from adaptive proofs, but the separation has several shortcommings: i) it relies on an inefficient interactive assumption, ii) it is specific to a proof system for a specific problem (discrete logarithm), and iii) it is specific to a class of proofs (those obtained from Σ-protocols via the Fiat-Shamir transform). That is, it does not pinpoint precisely the source of the difficulty and, in particular, it leaves open the question whether any adaptive proofs exist that are not also straight-line[2].

Our contribution. In this paper we obtain a full characterization of adaptive proofs in the random oracle model and we leverage this result to provide more general results regarding the limitations of the Fiat-Shamir transform. Below we outline our contributions.

[1] Recall that in the random oracle model hash functions can only be computed by calling an oracle available to all parties (and which on a fresh input returns a truly random output).

[2] Straight-line proofs are trivially adaptively secure.

ADAPTIVE PROOFS ≡ PROOFS WITH STRAIGHTLINE EXTRACTORS. Our main contribution is a negative answer to Bernhard et al.'s open question. It holds for all non-interactive proof schemes in the ROM, whether or not they are derived from Σ-protocols:

Theorem 1 (Informal). *Consider an arbitrary non-interactive proof of knowledge system in the ROM. If the proof system has an efficient adaptive extractor against adaptive provers then it also has a straight-line extractor.*

The immediate consequence of this theorem is that when designing PoKs for an adaptive setting, one cannot rely on rewinding and instead one should ensure the existence of a straightline extractor. While a general strategy is to employ Fischlin's transformation [11], one may still want to rely on the more efficient construction that uses the Fiat-Shamir transformation whenever possible – the impossibility result of Bernhard et al. [5] only applies to Fiat-Shamir-Schnorr proofs.

LIMITATIONS OF THE FIAT-SHAMIR TRANSFORM. We show that the Fiat-Shamir transformation has intrinsic limitations. In particular, we generalize the results of [5] in two distinct directions. On the one hand, we show that it holds for arbitrary Σ-protocols for proving knowledge of preimages of linear functions (including Schnorr, Chaum–Pedersen and representation proofs). More interestingly, we weaken the condition under which these proofs are not adaptively secure from one-more discrete logarithm (resp. one-more one-wayness [1]) to the following assumption: a dishonest verifier in a single execution of the Σ-protocol cannot extract the witness. We call this assumption *Σ-one-wayness*. Our result thus improves from a "q-type" assumption, which does not admit an efficient game, to an efficient game with only three rounds.

This theorem answers the main open question of [5] and hints at a shortcoming of proofs based on the Fiat–Shamir transform: if used in a setting where the prover gets to adaptively chosen statements, an extractor would have to be (more or less) straight-line. However, if this is the case, the proof may not be that interesting anyway as the underlying witness is not well-protected:

Theorem 2 (Informal). *Suppose there is a straight-line extractor for a Fiat-Shamir transformed Σ-protocol Σ (to prove knowledge of a preimage of some linear function f). Then a dishonest verifier can extract a witness (a preimage under f) in a single run of Σ.*

Taken together, these results imply that Fiat-Shamir transformed Σ-protocols are not adaptively secure in any setting in which they might be useful. Take the Schnorr protocol as an example: if Fiat-Shamir-Schnorr is adaptively secure then either discrete logarithms are easy in the relevant group, in which case the Schnorr protocol is redundant, or the Schnorr protocol provably helps a dishonest verifier to extract the discrete log of the statement — in which case Schnorr is certainly not zero-knowledge.

The theorem of Bernhard et al. [5] contains an adaptive prover who makes a sequence of n proofs such that each one depends on all previous ones. The

straightforward rewinding strategy — rewind on every proof to extract — ends up rewinding 2^n times since the rewound provers make new proofs which again have to be rewound. A combinatorial argument then shows that any strategy that rewinds fewer than 2^n times must have taken a discrete logarithm to find out the witness for one of the proofs output by the prover. The problem is that we do not know where, and also if we inject a challenge in one proof then we end up having to simulate all other proofs in the experiment. So far the solution to this problem was to reduce to the one-more discrete logarithm problem.

Our proof technique for Theorem 2 (formally, Theorem 14) is to take any prover and turn it into an adaptive prover who makes a chain of n proofs, together with some bookkeeping. We then show that any adaptive extractor against this prover must either take exponential time or reduce to a straight-line extractor against the original prover. Applying this theorem to the honest prover, we get a reduction to Σ-one-wayness. We summarize these results in the following table (FSS=Fiat-Shamir-Schnorr, DLOG=discrete logarithm, OMDL=one-more discrete logarithm).

property	breaks if FSS has
one-way (e.g. DLOG)	straight line extractor [18]
Σ-one-way	adaptive extractor (new)
one-more one-way (e.g. OMDL)	adaptive extractor [5]

In conclusion, we suggest that adaptive proofs are not a new class of proofs but rather another description of the class of straight-line extractable proofs; and Fiat-Shamir transformed Σ-protocols for "useful" functions are *not* in this class.

Weaker assumptions. The obvious next question is whether one can improve our results even further and only rely on one-wayness (e.g. in the case of Schnorr, DLOG) rather than Σ-one-wayness. We show using a meta-metareduction that no algebraic metareduction [16] from a programming extractor to one-wayness (e.g. DLOG) can exist, unless one-wayness is already easy. All previous metareductions in this area [18, 5] including ours to Σ-one-wayness are algebraic: the only operations they perform on elements of the target group are group operations.

A generalization of the Generic Group Model (GGM). To strengthen trust in the Σ-one-wayness hypothesis on which our impossibility relies we provide a justification in the generic group model. Interestingly, the first problem that one needs to face here is that the existing approaches to formalizing and using GGM is not suitable: in brief, an adversary that interacts with the Σ protocol for some problem gets to see not only group elements but also some information related to the exponents of these group elements – this ability is not considered the standard GGM formalizations. We suggest one approach to deal with this issue and use the resulting model to formally justify Σ-one-wayness.

Related work. One-more type assumptions were introduced by Bellare et al. [1]. Their value in proving schemes secure is subject to some debate, as explained by Koblitz and Menezes [13] who also gave the first "weakened" one-more assumption. Problems with forking-based proofs were first noted by Shoup and Gennaro [20]; Paillier and Vergnaud [16] developed separation results for Schnorr-based signatures using metareductions that formed the first formal proof of a limitation of Schnorr-based techniques. Both Brown [8] and Bresson et al. [9] concurrently applied separation techniques to one-more problems. Fischlin and Fleischhacker [12] were the first to consider limitations of metareductions via meta-metareductions. The most recent results that motivated this paper are Seurin and Treger [18] who gave a very simple metareduction from a non-programming extractor for Schnorr proofs to discrete log; and Bernhard et al. [5] who introduced adaptive proofs.

2 Preliminaries

NOTATION. $f : A \to B$ is a function with domain A and range B; $\mathcal{A} : A \twoheadrightarrow B$ is a randomised algorithm on the same domain and range. We write security games in a language based on Bellare and Rogaway's code-based game-playing [2]. $y \leftarrow f(x)$ is assignment, $x \twoheadleftarrow R$ is uniform random sampling. $T[i]$ is the element at index i of table T.

An interactive, randomised algorithm $\mathcal{A}$ has access to a random string r and an input/output interface. It maintains its state between calls. A security game is such an algorithm that may at some point output "win" or "lose", which terminates the entire execution. We say that a security property is given by a game, to mean that the property holds if no efficient adversary can cause the game to output "win" with more than negligible probability in some underlying security parameter.

Σ-PROTOCOLS — Let k be a field. Let $\mathcal{W}, \mathcal{X}$ be k-vector spaces and let $\phi : \mathcal{W} \to \mathcal{X}$ be a k-linear map, (i.e. linear map w.r.t. the field k). Suppose further that one can sample uniformly from k and $\mathcal{W}$.

Definition 3. *The Σ-protocol Σ_ϕ is the following interactive protocol for a prover P to prove knowledge of a preimage under ϕ to a verifier V.*

$$
\begin{array}{rcl}
P(w \in \mathcal{W}, x = \phi(w)) & & V \\
r \twoheadleftarrow \mathcal{W}; a \leftarrow \phi(r) & \xrightarrow{(x,a)} & \\
 & \xleftarrow{c} & c \twoheadleftarrow k \\
s \leftarrow r + c \cdot w & \xrightarrow{s} & \phi(s) \stackrel{?}{=} a + c \cdot x
\end{array}
$$

We choose to let P transmit the statement x to V as part of the first round of the proof — of course, V may also know x in advance. The verifier accepts if the equation $\phi(s) = a + c \cdot x$ holds in $\mathcal{X}$, in which case we call (x, a, c, s) an accepting transcript. Instances of this template protocol include:

- Schnorr: $\mathcal{W} = k = GF(p)$, $\mathcal{X}$ is some group G of order p with a generator g (e.g. over an elliptic curve) and $\phi(w) = g^w$.
- Chaum-Pedersen: $\mathcal{W} = k$, $\mathcal{X} = G \times G$ for a group as above and $\phi(w) = (g^w, h^w)$ for two different generators g, h of G.
- Representation: $\mathcal{W} = k^n, \mathcal{X} = G$ and $\phi(w_1, \ldots, w_n) = \prod_{i=1}^{n} {g_i}^{w_i}$ for some known set of generators $\{g_i\}_{i \in I}$ of G.

Σ-protocols according to our definition automatically satisfy:

- Special soundness: if (x, a, c, s) and (x, a, c', s') are accepting transcripts with $c \neq c'$ then $1/(c - c') \cdot (s - s')$ is a preimage[3] of x under ϕ.
- Soundness: if $x' \in \mathcal{X} \setminus \mathrm{Im}[\phi]$, then a cheating prover gets a verifier to accept with probability at most $1/|k|$.
- Honest-verifier zero-knowledge: a verifier who chooses c as prescribed (at least, independently of a) gains no information from the protocol beyond the fact that the prover knows a preimage of x under ϕ.

Proof schemes — A (non-interactive) proof scheme for a relation ρ on sets X, W consists of a proof space Π and a pair of algorithms $\texttt{prove} : X \times W \twoheadrightarrow \Pi$ and $\texttt{verify} : X \times \Pi \to \{0, 1\}$ (i.e. `verify` is deterministic). An element $\pi \in \Pi$ satisfying $\texttt{verify}(x, \pi) = 1$ is called a valid proof for x. For any (x, w) satisfying ρ, if $\pi \leftarrow \texttt{prove}(x, w)$ then we require $\texttt{verify}(x, \pi) = 1$. We further assume that there is an algorithm $\texttt{sample} :\twoheadrightarrow X \times W$ that produces elements uniformly distributed in ρ (as a subset of $X \times W$). In the random oracle model (ROM), both `prove` and `verify` may call a function H that is modelled as a random oracle in security proofs. The relation ρ itself does not use H.

Fiat-Shamir — The Fiat-Shamir transformation turns Σ-protocols into non-interactive proof schemes that are full zero-knowledge proofs of knowledge in the random oracle model. The idea is simply to replace the verifier's challenge c by a hash over the statement x and the commitment a.

Definition 4. *Let $\phi : \mathcal{W} \to \mathcal{X}$ be a k-linear function where $\mathcal{W}$ and k are efficiently sampleable. Suppose that H is a function with domain (including) $\mathcal{X} \times \mathcal{X}$ and range k. Then the Fiat-Shamir transformed Σ-protocol $\mathcal{F}_\phi$ is the following proof scheme for sets $(\mathcal{X}, \mathcal{W})$ and relation $\rho(x, w) = 1 \iff \phi(w) = x$. The proof space is $\Pi = \mathcal{X} \times \mathcal{W}$ and the algorithms are*

- `sample()`*: pick $w \twoheadleftarrow \mathcal{W}$, set $x \leftarrow \phi(w)$ and return (x, w).*
- `prove`(x, w)*: pick $r \twoheadleftarrow \mathcal{W}$, set $a \leftarrow \phi(r)$, $c \leftarrow H(x, a)$ and $s \leftarrow r + cw$. The proof is $\pi = (a, s)$.*
- `verify`$(x, (a, s))$*: check that $\phi(s) = a + H(x, a) \cdot x$.*

[3] The inversion $1/(c - c')$ is in the field k where it exists due to $c \neq c'$; the dot in this formula is field-vector multiplication.

3 Variations on the theme of one-wayness

Σ-protocols are only useful when the function ϕ is hard to invert: otherwise, they are trivially zero-knowledge proofs of knowledge, but so is the protocol in which the prover just sends the statement x to the verifier. For the same reasons, if ϕ is easy to invert then $\mathcal{F}_\phi$ is adaptively secure too. This shows that we cannot hope for a theorem of the form "Fiat-Shamir Schnorr is not adaptively secure", since it is adaptively secure e.g. in the group $(\mathbb{Z}_p, +)$ where taking discrete logarithms is easy. Consequently, limitation theorems take the form "if Schnorr is adaptively secure then some property (e.g. OMDL) is easy to break".

We discuss some possible security properties of the function ϕ and introduce Σ-one-wayness. Later on we will show that Fiat-Shamir proofs cannot be adaptively secure unless Σ-one-wayness of ϕ is easy to break (in which case, their use within protocols would be already questionable).

Σ-ONE-WAYNESS. One-wayness is the first obvious candidate property. Recall that a property defined by a game means that the property (in this case one-wayness of ϕ) holds if it is hard to make the game output "win". The game gives the adversary a uniformly chosen image x; the adversary wins by recovering any preimage w' s.t. $\phi(w') = x$.

Definition 5. *The one-wayness property for a function $\phi : \mathcal{W} \to \mathcal{X}$ is given by the following game.*

1 $w \twoheadleftarrow \mathcal{W}; x \leftarrow \phi(w)$
2 `output` x`;` `input` $w' \in \mathcal{W}$
3 `if` $\phi(w') = x$ `then return` "win" `else return` "lose" `end`

We propose a new security notion that we call Σ-one-wayness (for linear functions). This says that even a dishonest verifier (who chooses c arbitrarily, maybe depending on x and a) cannot extract w from a single run of the protocol. We do not claim that Σ-one-wayness is a sufficient security notion for Σ protocols (it says nothing about extracting partial information on w) but we postulate that it is only deployed if this condition is satisfied. Consequently, we are not proposing a new scheme that is secure under the Σ-one-wayness assumption, but we will claim that if the Fiat-Shamir proof scheme for a function ϕ is adaptively secure then the Σ-one-wayness property for ϕ is easy to break too. Σ-one-wayness is clearly stronger than one-wayness of the function ϕ, but it will turn out to be weaker than one-more one-wayness (which we define in a moment).

Definition 6. *The Σ-one-wayness property for a linear function $\phi : \mathcal{W} \to \mathcal{X}$ is defined by the following game.*

```
1 | w ↞ 𝒲; x ← φ(w)
2 | r ↞ 𝒲; a ← φ(r)
3 | output (x, a); input c ∈ k
4 | s ← r + c · w
5 | output s; input w′ ∈ 𝒲
6 | if φ(w′) = x then return "win" else return "lose" end
```

If Σ-one-wayness of a function ϕ is easy to break but one-wayness is hard to break, then running the protocol Σ_ϕ could leak a preimage to a dishonest verifier, that said verifier could otherwise not compute by herself. In this case we would discourage the use of the protocol Σ_ϕ (among other things it is certainly not zero-knowledge). If both Σ-one-wayness and one-wayness of ϕ are easy to break then Σ_ϕ is both harmless and useless. So, we think that Σ-one-wayness of ϕ is a necessary condition for the protocol Σ_ϕ to be deployed. Under this condition, we will show that the Fiat-Shamir transformed Σ_ϕ is not adaptively secure.

Weak one-more one-wayness. For Schnorr, the one-wayness property of the function $\phi(x) = g^x$ is the discrete logarithm property. Bernhard et al. [5] use a stronger assumption known as one-more discrete logarithm, which generalises to one-more one-wayness[1, 9].

One-more one-wayness assumes an invertible function ϕ. The adversary is given two oracles which she can call in any order: a sampling oracle that picks a random preimage $w_i \in \mathcal{W}$ and reveals $x_i = \phi(w_i)$ and an opening oracle that on input x outputs $\phi^{-1}(x)$. To win the game, the adversary must recover the preimages of all samples x_i with fewer calls to the opening oracle than to the sampling oracle.

One-more one-wayness was first discussed by Bellare et al. [1] although the name first appears in a later paper [9]. Unlike the other properties here, it does not admit an efficient security game: the game itself needs to be able to invert ϕ on arbitrary inputs. Koblitz and Menezes [13] discussed a variant that only allows the adversary to open challenges themselves; this is insufficient for our applications. Instead we propose a new property: weak one-more one-wayness fixes this problem by restricting the adversary to asking linear combinations of the sampled challenges. Your task is still to recover all w_i with fewer queries to the linear combination oracle than to the sampling oracle. The requirement for ϕ to be invertible can also be dropped again.

Definition 7. *The weak and normal one-more one-wayness properties for a bijection $\phi : \mathcal{W} \to \mathcal{X}$ are given by the following games. In each game the adversary can call the* `sample` *and* `open` *oracles many times, in any order. The adversary wins the game if it can provide preimages under ϕ for all samples that it had obtained and yet it made fewer opening queries than sample queries.*

weak one-more one-wayness:

```
1 sample():
2     n ← n + 1
3     w[n] ↞ W
4     return φ(w[n])
5
6 open(c_1, ..., c_n ∈ k^n):
7     return Σ_{i=1}^{n} c_i · w_i
```

one-more one-wayness:

```
1 sample():
2     (same as weak version)
3
4
5
6 open(x ∈ X):
7     return φ^{-1}(x)
```

The strong problem clearly reduces to the weak one. A weak adversary can still obtain a preimage of a particular sample by submitting the vector with 1 at the appropriate position and 0 elsewhere.

The point of the weak one-more one-wayness property is that the theorem by Bernhard et al. [5] can trivially be strengthened to show that Fiat-Shamir-Schnorr is not adaptively secure even under the weak one-more discrete logarithm property: their reduction only ever makes opening queries on elements that are linear combinations of samples with known coefficients. This is not surprising since their reduction is trying to "simulate" Schnorr proofs on sample elements.

Σ-one-wayness reduces to weak one-more one-wayness, even to a weaker version where the number of samples is additionally bounded at 2 and only a single linear combination query is allowed. Thus, we end up with a hierarchy of one-wayness / Σ-one-wayness / weak one-more one-wayness / one-more one-wayness, in order of increasing strength.

4 Adaptive proofs

In this section we recall the notion of adaptive proofs and introduce templates for a couple of provers that form the basis of the results we prove in the next section.

Definition 8. *A prover is an algorithm $\mathcal{P}$ that outputs a statement/proof pair (x, π); in the ROM a prover may make random oracle calls. We assume that there is a uniformly sampleable space of random strings R associated to each prover and we write $\mathcal{P}(r)$ to mean running prover $\mathcal{P}$ on random string $r \in R$. In the ROM, we write $(x, \pi, l) \leftarrow \mathcal{P}(r)$ to mean that we also return the list l of random oracle queries made by this execution of the prover on random string r.*

A proof scheme is sound with error ε if for any prover $\mathcal{P}$, the probability of producing a pair (x, π) such that $\texttt{verify}(x, \pi) = 1$ but no w exists making $\rho(x, w)$ hold, is at most ε.

A proof scheme in the random oracle model is straight-line extractable with error ε if there is an extractor $\mathcal{K}$ as follows. For any prover $\mathcal{P}$, pick $r \twoheadleftarrow R$ and execute $(x, \pi, l) \leftarrow \mathcal{P}(r)$. If $\texttt{verify}(x, \pi) = 1$ w.r.t.[4] l then with probability at least $1 - \varepsilon$, $\mathcal{K}(x, \pi, l)$ returns a w such that $\rho(x, w)$ holds. It follows immediately

that an extractable proof scheme with error ε is also sound with at most the same error.

A straight-line extractor, has black-box access to further copies of the prover in the following sense: it may start these copies and control all interaction with them, including picking the random string and answering all random oracle queries. As motivation for this (established) notion of straightline extractors, consider an extractor who is trying to extract from an honest prover. The code of the honest prover is known, so the extractor can always simulate the honest prover on inputs of its choice. The extractor cannot see the random coins of the "main" copy of the honest prover from which it is trying to extract, however.

A proof scheme in the ROM has a *programming* straight-line extractor with error ε if there is a straight-line extractor $\mathcal{K}$ as follows. Let any prover $\mathcal{P}$ interact with $\mathcal{K}$ in the sense that $\mathcal{K}$ answers $\mathcal{P}$'s random oracle queries. If $\mathcal{P}$ outputs (x, π) such that $\texttt{verify}(x, \pi) = 1$ w.r.t. the oracle queries made by $\mathcal{P}$, then with probability at least $1 - \varepsilon$, $\mathcal{K}$ outputs a w such that $\rho(x, w)$ holds.

A rewinding extractor can, in addition to the capabilities of a straight-line extractor, launch further copies of the prover $\mathcal{P}$ with the *same random string* as the one that the extractor is trying to extract from, and answer their random oracle queries. The difference between straight-line and rewinding extractors is thus that rewinding extractors can run further copies of the prover that behave identically to the "main" one, as long as they receive the same inputs and outputs, and thus "fork" the prover instance from which they are trying to extract.

Adaptive proofs. We present the adaptive proof game of Bernhard et al. [5]. The game is an interactive algorithm with two interfaces for an adaptive prover and an extractor. An adaptive prover for a proof scheme $(\Pi, \texttt{prove}, \texttt{verify})$ w.r.t. (X, W, ρ) is an algorithm that can repeatedly output pairs $(x, \pi) \in X \times \Pi$ and expect witnesses $w \in W$ in return. After a number of such interactions, the adaptive prover halts. In the random oracle model, an adaptive prover may also ask random oracle queries.

An adaptive extractor is an algorithm $\mathcal{K}$ that can interact with an adaptive prover: it repeatedly takes pairs $(x, \pi) \in X \times \Pi$ such that $\texttt{verify}(x, \pi) = 1$ as input and returns witnesses $w \in W$ such that $\rho(x, w) = 1$. In addition, an adaptive extractor may be rewinding, i.e. launch further copies of the adaptive prover that run on the same random string as the main one (managed by the game) and answer all their queries. The adaptive proof game does not check the correctness of witnesses for the other copies of the prover.

Definition 9. *A proof scheme $(\Pi, \texttt{prove}, \texttt{verify})$ is an n-proof (with error ε) in the random oracle model if there is an adaptive extractor $\mathcal{K}$ such that for any adaptive prover $\mathcal{P}$, the extractor wins the game in Figure 1 (with probability at least $1 - \varepsilon$). The adaptive extractor $\mathcal{K}$ may launch and interact with further*

[4] We say $\texttt{verify}(x, \pi) = 1$ w.r.t. l if all elements on which verify queries the oracle on input (x, π) are contained in l, and verify outputs 1 on these inputs if given the appropriate responses in l.

```
1  | initialise:
2  |     Q ← [ ]
3  |     K ← 0
4  |     r ↞ R
5  |     run P(r)
6  |
7  | P asks ro(x):
8  |     y ← ro(x)
9  |     Q ← Q :: (x, y)
10 |     send y to P
11 |
12 | P outputs (x, π):
13 |     if not verify(x, π) then
14 |         K wins; halt.
15 |     end
16 |     Ξ ← x
17 |     send (x, π, Q) to K
18 | K asks ro(x):
19 |     y ← ro(x)
20 |     send y to K
21 |
22 | K outputs w:
23 |     if ρ(Ξ, w) then
24 |         K ← K + 1
25 |         if K = n then
26 |             K wins; halt.
27 |         end
28 |         send w to P
29 |     else
30 |         P wins; halt.
31 |     end
32 | P  halts:
33 |     K wins; halt.
```

Fig. 1. The adaptive proof game. The extractor also has access to further copies of $\mathcal{P}(r)$ – to avoid cluttered notation we do not show this access explicitly.

copies of the adaptive prover $\mathcal{P}$ on the same random string r as the main one, without the game mediating between them.

In the n-proof game in Figure 1, the prover $\mathcal{P}$ is trying to find a claim (x, π) that verifies, but from which the extractor $\mathcal{K}$ cannot extract a witness w. The extractor is trying to extract witnesses from all claims made by the prover.

The game uses three global variables. K stores the number of witnesses that the extractor has found so far. If this counter reaches n, the extractor wins and the scheme is an n-proof. Q stores a list of all the prover's random oracle queries so far. These are provided[5] to the extractor along with each of the prover's claims. Ξ stores the last statement that appeared in one of the prover's claims; it is used to check the validity of a witness returned by the extractor.

A n-proof for $n = 1$ is simply a proof of knowledge in the ROM: the prover makes a single claim (a pair containing a statement x and a proof π) and the extractor wins if it can obtain a witness. A proof scheme is an adaptive proof if there is an adaptive extractor that works for any polynomially bounded parameter n.

Canonical provers. The canonical prover $\mathcal{P}_C$ samples a statement/witness pair and creates a proof. We write R_C for the randomness space of the canonical prover and $\mathcal{P}_C(r)$ to denote running the canonical prover on the random string $r \in R_C$.

[5] The original definition gave the extractor an extra `list` oracle to query the prover's random oracle list. Our presentation is equivalent.

$\mathcal{P}^n(r)$:

```
1  for i = 1, n do
2      (x_i, π_i, l) ← P(r)
3      r ← F(r, l)
4  end
5  for i = n, 1, -1 do
6      output (x_i, π_i)
7      input w'
8      if not ρ(x_i, w') then
9          halt
10     end
11 end
```

Fig. 2. The adaptive chain prover $\mathcal{P}^n$. The provder depends on function F which here we assumed fixed. When F is a pseudorandom function, a key for F is sampled at the beginning of the execution of $\mathcal{P}^n$.

Definition 10 (canonical prover). *Let* $(\Pi, \texttt{prove}, \texttt{verify})$ *be a proof scheme for* (X, W, ρ) *where* r *is uniformly sampleable via an algorithm* `sample`*. The canonical prover* $\mathcal{P}_C$ *for this scheme is the following algorithm.*

$$(x, w) \leftarrow \texttt{sample()};\ \pi \leftarrow \texttt{prove}(x, w);\ \texttt{return}\ (x, \pi)$$

If required, the canonical prover can also return the list l *of all random oracle queries made during its execution (by* `prove`*).*

Since an extractor is supposed to work against any (efficient) prover, to argue that an extractor cannot exist it is enough to show that one cannot extract from the canonical prover. To deal with adaptive extractors, we propose the following construction of an adaptive chain prover $\mathcal{P}^n$ from any prover $\mathcal{P}$. It follows the idea of Shoup and Gennaro [20] in making a chain of "challenges" (in this case proofs) where each challenge depends on all previous ones and then asking queries on them *in reverse order*. This way, the obvious rewinding extractor using special soundness will take exponential time.

To make each challenge depend on previous ones, we use a function F to update the random string for each of the n contained copies of $\mathcal{P}$ based on the (random oracle) state of the previous copy. The final parameter l returned by $\mathcal{P}$ is the list of all random oracle queries made by this copy. Recall that $\mathcal{P}(r)$ means run prover $\mathcal{P}$ on random string $r \in R$ where R is the randomness space for this prover.

Definition 11 (adaptive chain prover). *Let* $(\Pi, \texttt{prove}, \texttt{verify})$ *be a proof scheme for* (X, W, ρ)*. Let* $\mathcal{P}$ *be any prover (in the ROM) and let* R *be its randomness space. Let* L *be the space of possible random oracle input/output transcripts. Let* $F : R \times L \to R$ *be a function (which does not depend on the random oracle). The adaptive chain prover* $\mathcal{P}^n$ *of order* n *w.r.t. function* F *is the algorithm in Figure 2, taking an* $r \in R$ *as input.*

Later on, we will take F to be a (pseudo-)random function. This has the effect that two copies of $\mathcal{P}^n$ that get identical answers to their random oracle queries will behave identically, but two copies of $\mathcal{P}^n$ that "fork" will behave as copies of $\mathcal{P}$ with *independent* random strings from the forking point onwards.

The intuition behind this construction is that having access to copies of $\mathcal{P}$ on some uniformly random string r' cannot help you extract from a copy $\mathcal{P}(r)$, as long as r and r' are independent — certainly, an extractor could always simulate such copies herself if the code of $\mathcal{P}$ is known. We will use this idea to show that forking a copy of $\mathcal{P}^n$ is no help in extracting from the proofs made later on by another copy.

5 Limitations of the Fiat-Shamir transformation

We recall the hierarchy of security definitions for functions (the last is the strongest): one-way / Σ-one-way / weak one-more one-way / one-more one-way.

KNOWN LIMITATIONS. Seurin and Treger [18] proved that Fiat-Shamir-Schnorr cannot have a non-programming straight-line extractor unless the underlying function is not one-way (i.e. one can take discrete logarithms). The following theorem generalizes this result to the case of arbitrary Σ-protocols.

Theorem 12. *Suppose there is a non-programming straight-line extractor K for the proof scheme $\mathcal{F}_\phi$. Then ϕ is not one-way. Specifically, there is an algorithm breaking one-wayness with approximately the same running time and success probability as the extractor K has against the canonical prover $\mathcal{P}_C$.*

Proof. Let x be a challenge from the one-way game for ϕ; we need to find a w' such that $\phi(w') = x$. We simulate a proof: pick $s \twoheadleftarrow \mathcal{W}$, $c \twoheadleftarrow k$ and $a \leftarrow \phi(s) - c \cdot x$. Then we give the extractor the statement x, proof (a, s) and a list of random oracle queries consisting of the entry $RO(x, a) = c$. These elements are identically distributed to what the extractor would see in an execution with the canonical prover $\mathcal{P}_C$ for $\mathcal{F}_\phi$. We pass any witness w' returned by the extractor on to the challenger to win with the same success probability. □

Bernhard et al. [5] showed that substituting an adaptive extractor for a straight-line one gets a similar result for the one-more one-wayness assumption on the function ϕ — the proof of this theorem is nontrivial however. While their original proof only concerned Fiat-Shamir-Schnorr, a close inspection of the proof shows that it works for any Σ protocol and that the weak one-more assumption is sufficient too. In summary:

Theorem 13. *Suppose that there is an efficient adaptive extractor for $\mathcal{F}_\phi$. Then ϕ is not weak one-more one-way.*

We will not re-prove the theorem here. As to running time, as long as the extractor makes fewer than 2^n queries when running against a particular prover then the reduction to weak one-more one-wayness runs in the same time as the extractor, but its success probability is the inverse of the number of copies of the prover that the extractor causes to be invoked. Although Bernhard et al. [5] only prove the theorem for the case of the Fiat-Shamir-Schnorr protocol, their reduction is "black box" in the sense that it only needs to be able to sample and

open instances of the underlying Σ-protocol. The exact same proof will work for our generalisation. We can write the prover in the cited theorem as $(\mathcal{P}_C)^n$, the adaptive chain prover derived from the canonical prover. This will allow us to conclude that any extractor against the chain prover implies a straight-line extractor against the canonical prover $\mathcal{P}_C$.

New Results. If we switch to a programming straight-line extractor, we can show a separation result under the Σ-one-wayness assumption.

Theorem 14. *Suppose there is a programming straight-line extractor K for $\mathcal{F}_\phi$. Then ϕ is not Σ-one-way. Specifically, there exists a reduction with approximately the same running time and the same success probability as the extractor K against the canonical prover $\mathcal{P}_C$.*

Proof. We simulate the canonical prover towards the extractor. Receive x, a from the Σ-one-way challenger and ask the random oracle query $c \leftarrow RO(x, a)$ (which the extractor answers). Then send c to the Σ-one-way challenger to get s and send (x, a, s) to the extractor. Again, whenever the extractor provides the correct w', we win against the challenger. □

This result is new, but not surprising — Fiat-Shamir transformed Σ-protocols are not supposed to be straight-line extractable and the Σ-one-wayness property is constructed exactly to make this reduction work. The value of said property is that we can also use it for adaptive extractors.

Our main contribution in this paper is a new theorem that says all adaptive proofs in the ROM admit a straight-line extractor.

Theorem 15. *Consider any non-interactive ROM proof scheme with an adaptive extractor $\mathcal{K}$. Suppose that, running against any n-prover $\widehat{\mathcal{P}}$, the extractor $\mathcal{K}$ causes at most $f(n) < 2^n$ copies of the prover to be run in the experiment and answers all extraction queries of the main run correctly with probability at least $p(n) > 0$. Then there is a programming straight-line extractor against any non-adaptive prover $\mathcal{P}$ with success probability $p(n)/(n \cdot f(n))$.*

Remark 16. The proof, which we provide in the full version, is information theoretic. It relies only on the number of copies of P instantiated by $\mathcal{K}$ and, in particular, it makes no assumptions on the efficiency of $\mathcal{P}$ and $\mathcal{K}$. It does not establish a relationship between the running time of $\mathcal{K}$ and the success probability; it is simply an application of the pigeonhole principle to derive a contradiction that whenever $f(n) < 2^n$ then $\mathcal{K}$ must essentially have "guessed correctly" rather than computed the preimage through interacting with $\mathcal{P}$. For example, if launching a new copy of $\mathcal{P}$ costs $\mathcal{K}$ one unit of time then the theorem provides negative results even for subexponential-time extractors.

Applying this theorem to protocols obtained via the Fiat-Shamir transform from Σ-protocols yields the following insight.

Corollary 17. *Suppose that the Fiat-Shamir transformed Σ-protocol $\mathcal{F}_\phi$ is adaptively secure. Then ϕ is not Σ-one-way secure.*

For the corollary, note that we have a programming straight-line extractor against the canonical prover $\mathcal{P}_C$ by applying Theorem 15 to the prover $(\mathcal{P}_C)^n$. The result then follows from Theorem 14.

We sketch the proof here and provide the full argument in the full version of this paper [7]. Let $\mathcal{P}$ be a non-adaptive prover. We construct a simulator $\mathcal{S}^n$ that is indistinguishable from the black box providing "rewinding" access for multiple copies of $\mathcal{P}^n$. The point of the simulator is that it shares state "between the copies". We then guess which instance of the prover (specifically, which proof) the extractor is going to answer without "forking" and inject the Σ-one-wayness challenge into it. The same combinatorial argument as in the proof of Bernhard et al. [5] shows that such an instance must exist if the extractor launches fewer than 2^n copies of the prover.

The core of such a simulation argument is to keep track of a history of each instance of the prover, since two copies of the prover with identical histories must behave identically towards the extractor. In $(\mathcal{P}_C)^n$, this history is implicitly tracked in the randomness r used for each copy of $\mathcal{P}_C$, however a collision in the random oracle could lead to two copies with different histories "merging". Our simulator computes an explicit history instead, namely the list of all random oracle queries so far.

As in Bernhard et al. [5] we define an event E that occurs whenever a copy of the prover gets its extraction query answered without having a "partner" (another copy, from which the witness was extracted by forking and special soundness). The novelty in our proof is that because we have cast the prover as a chain $(\mathcal{P}_C)^n$ with suitable state tracking, we can show that event E implies not only a "break" of the chain prover but also of one of the contained canonical provers $\mathcal{P}_C$. We then show that, if the simulator guessed correctly, event E implies that the simulator can solve its Σ-one-wayness challenge. This is a much weaker assumption than one-more one-wayness.

6 Generic hardness of Σ-one-wayness

In this section we show that Fiat-Shamir transformed Σ-protocol $\mathcal{F}_\phi$ is not adaptively secure in the generic setting. Thus if we want to build a protocol where we need an adaptively secure proof (such as to get CCA encryption), we would not use $\mathcal{F}_\phi$. By Corollary 17 we just need to prove that ϕ is $\Sigma-$one-way secure in the generic group model (GGM). Again, let $\mathcal{X}$, $\mathcal{W}$ be vector spaces over k and $\phi : \mathcal{W} \to \mathcal{X}$ be a k-linear map. We assume that k is a finite field and $\mathcal{X}, \mathcal{W}$ are finite dimensional, since sampling uniformly from an infinite set does not make much sense. Let $b_1, b_2, ..., b_n$ be basis vectors of $\mathcal{X}$, where $\dim(\mathcal{X}) = n$ and $\{b_1, ..., b_s\}$ be a basis for $\mathrm{Im}(\phi)$. For each $1 \leq i \leq s$ denote a_i to be an element of $\mathcal{W}$ so that $\phi(a_i) = b_i$.

The generic group model is a model which analyses success of algorithms against representations of groups which do not reveal any information to adver-

sary. There are many ways to formalise this idea [14, 15, 19]. We will follow the definition provided by Shoup [19]. Here, adversary is given access to images of elements of a group under a random injective map $\sigma : \mathcal{X} \to S \subset \{0,1\}^*$, called an *encoding function*. Group operations can be computed by making oracle queries. The adversary is given access to two oracles ADD and INV:

$$ADD(\sigma(x), \sigma(y)) = \sigma(x+y),\ INV(\sigma(x)) = \sigma(-x).$$

Note that the adversary cannot get any information from the representation $\sigma(x)$ of element x. A *generic algorithm* $\mathcal{A}$ for $\mathcal{X}$ on S is a probabilistic algorithm that takes as input an encoding list $(\sigma(x_1), \sigma(x_2), ..., \sigma(x_l))$ where each $x_i \in \mathcal{X}$ and σ is an encoding function of $\mathcal{X}$ on S. As the algorithm executes, it makes queries to ADD or INV oracles and then appends outputs of the queries to the encoding list. The output of $\mathcal{A}$ is a bit string denoted as $\mathcal{A}(\sigma; x_1, ..., x_l)$. We also want to extend the interface of the model and introduce the $FSS_{r,w}$ oracle: $FSS_{r,w}(c) = r + cw$ for some $r, w \in \mathcal{W}$ and $c \in k$. We may assume that when oracles encounter some $\sigma_1 \notin Im(\sigma)$, they return an error message.

The next theorem establishes generic hardness of $\Sigma-$one-wayness; we provide the proof in the full version of this paper [7].

Theorem 18. *Let w, r be random elements of $\mathcal{W}$ and $\mathcal{A}$ be a generic algorithm for $\mathcal{X}$ on $S \subset \{0,1\}^*$ that makes at most m queries to ADD and INV oracles and exactly one query to $FSS_{r,w}$ oracle. Then the probability that $\phi(\mathcal{A}(\sigma; b_1, ..., b_n, x, y)) = x$ is $O((m+n)^2/|\mathcal{X}| + |ker(\phi)|/|\mathcal{W}|)$, where $x = \phi(w)$ and $y = \phi(r)$.*

Note that Theorem 18 implies that every generic algorithm $\mathcal{A}$, which wins $\Sigma-$one-wayness with high probability, must perform at least $\Omega(\alpha\sqrt{|\mathcal{X}|})$ group operations, where $\alpha = \sqrt{1 - |\ker(\phi)|/|\mathcal{W}|}$. In particular, if $\mathcal{W}$ is large then we get the lower bound $\Omega(\sqrt{|\mathcal{X}|})$ for IES (described in [6]) by choosing $\phi(w) = g^w$.

7 Reducing to DLOG?

Given a non-programming straight line extractor in the ROM for a Fiat-Shamir transformed Σ-protocol $\mathcal{F}_\phi$ we can break one-wayness of ϕ; for a programming extractor or an adaptive extractor we can break Σ-one-wayness. This raises the question, can we break one-wayness given a programming extractor? Our answer is negative. We give the argument for the case of Schnorr proofs where one-wayness is the discrete logarithm (DLOG) problem; this also implies that there can be no generic metareduction to one-wayness for any Σ-protocol.

The metareductions in the theorems of Seurin and Treger [18], Bernhard et al. [5] and this paper are all algebraic (in the sense of Paillier and Vergnaud) [16] over the vector space[6] $\mathcal{X}$, the range of the function ϕ. We therefore consider it a meaningful result to show that no algebraic metareduction to DLOG can exist (unless DLOG is already easy).

[6] Paillier and Vergnaud defined the algebraic model for groups; one can interpret a $GF(p)$ vector space as an Abelian group to use their definition of the algebraic model.

Theorem 19. *If there is an algebraic metareduction from a programming straight-line extractor for Fiat-Shamir-Schnorr proofs to the DLOG problem then there is also a meta-metareduction breaking the DLOG problem directly with approximately the same success probability.*

The proof is in [7]. The idea is that a metareduction M gets to see two bases in the group: the generator g and the challenge h from its DLOG challenger. Since we assumed a programming extractor, M must ask its random oracle queries to its extractor interface where our meta-metareduction will answer them. Any statement output by M (to the extractor) therefore has the form $(g^a h^b)$ for some (a, b) which are available to our meta-metareduction by use of the algebraic model. Proofs of the form $(a, 0)$ are independent of the challenge h; intuitively they should not help to compute the discrete logarithm of h so we just return the witness a. The first time M outputs a proof with a statement of the form (a, b) with $b \neq 0$, we fork M on the relevant random oracle query and use special soundness to find the discrete logarithm of h to basis g.

8 Conclusions

Bernhard et al. introduced adaptive proofs, setting up a hierarchy of (1) proofs of knowledge (2) adaptive proofs and (3) straight-line extractable proofs, with a separation between (1) and (2). While useful for proving limitations of Σ-protocols, we have showed that adaptive proofs are not a new class of proof after all: all adaptively secure proofs admit a straight-line extractor against the canonical prover.

Along the way we have generalised previous results from Schnorr's protocol to Σ-protocols. In addition, we have weakened the counter-assumption from one-more one-wayness, which is a "q-type" interactive assumption (adversary gets an unbounded number of sample queries) and is not efficiently realisable to Σ-one-wayness, which both has a constant number of steps and an efficient security game.

Our result shows that the Fiat-Shamir transformation and Σ-protocols in general may be even weaker than previously thought. Namely, the non-interactive proof scheme $\mathcal{F}_\phi$ only achieves adaptive security if a single execution of the interactive protocol Σ_ϕ against a dishonest verifier already leaks the secret witness with non-negligible probability.

In essence, this shows that using proofs derived from the Fiat-Shamir scheme for some problem ϕ in a setting where adaptive security of such proofs is necessary requires care: these should be replaced with proofs that have straightline extractors. From a practice-oriented perspective, our results show that improving the efficiency of proofs that admit straightline extraction is an important line of future research.

References

1. M. Bellare, C. Namprempre, D. Pointcheval and M. Semanko. The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme. eprint 2001/002. Originally appeared as "The Power of RSA Inversion Oracles and the Security of Chaum's RSA-Based Blind Signature Scheme" in Financial Cryptography, LNCS 2339, pages 319–338, Springer 2001.
2. M. Bellare and P. Rogaway. Code-based game-playing proofs and the security of triple encryption. In: Advances in Cryptology — Eurocrypt '06, LNCS 4004, pages 409–426, 2006. The title cited is from the latest version on eprint at `http://eprint.iacr.org/2004/331`.
3. M. Bellare and P. Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In: CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, pages 62–73.
4. D. Bernhard. Zero-Knowledge Proofs in Theory and Practice. PhD thesis, University of Bristol, 2014. Available at `www.cs.bris.ac.uk/~bernhard/papers.html`
5. D. Bernhard, M. Fischlin and B. Warinschi. Adaptive Proofs of Knowledge in the Random Oracle Model. PKC '15, LNCS 9020, Springer, pages 629–649, 2015.
6. D. Bernhard, M. Fischlin and B. Warinschi. On the hardness of proving CCA-security of signed ElGamal. In Public-Key Cryptography-PKC 2016 (pp. 47-69). Springer Berlin Heidelberg.
7. D. Bernhard, N.K. Nguyen, B. Warinschi. Adaptive Proofs have Straightline Extractors (in the Random Oracle Model). Full version on eprint 2015/712.
8. D. Brown. Irreducibility to the One-More Evaluation Problems: More May Be Less. eprint 2007/435.
9. E. Bresson, J. Monnerat and D. Vergnaud. Separation Results on the "One-More" Computational Problems. In: CT-RSA '08, LNCS 4964, pages 71–87, Springer 2008.
10. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In: Proceedings on advances in cryptology (CRYPTO '86), pages 186–194, 1986.
11. M. Fischlin. Communication-Efficient Non-Interactive Proofs of Knowledge with Online Extractors. In: Proceedings of the 25th annual international cryptology conference on advances in cryptology (CRYPTO '05), pages 152–168, 2005.
12. M. Fischlin and N. Fleischhacker. Limitations of the Meta-Reduction Technique: The Case of Schnorr Signatures. In: Eurocrypt '13, LNCS 7881, pages 444–460, Springer 2013. eprint 2013/140.
13. N. Koblitz and A. Menezes. Another look at non-standard discrete log and Diffie-Hellman problems. In: Journal of Mathematical Cryptology, vol. 2 issue 4, pages 311–326, 2008. eprint 2007/442.
14. U. Maurer. Abstract models of computation in cryptography. IMA International Conference on Cryptography and Coding. Springer Berlin Heidelberg, 2005.
15. V.I. Nechaev. Complexity of a determinate algorithm for the discrete logarithm. Mathematical Notes 55.2 (1994): 165-172.
16. P. Paillier and D. Vergnaud. Discrete-Log Based Signatures may not be Equivalent to Discrete Log. In: Asiacrypt '05, LNCS 3788, pages 1–20, Springer 2005.

17. C.P. Schnorr. Efficient signature generation for smart cards. In: Journal of cryptology, Volume 4, Springer, pages 161–174, 1991.
18. Y. Seurin and J. Treger. A Robust and Plaintext-Aware Variant of Signed ElGamal Encryption. In: CT-RSA LNCS 7779, pages 68–83, Springer 2013.
19. V. Shoup Lower bounds for discrete logarithms and related problems. International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 1997.
20. V. Shoup and R. Gennaro. Securing Threshold Cryptosystems Against Chosen-Ciphertext Attack. In: Advances in Cryptology (Eurocrypt '98), LNCS 1403, pages 1–16, 1998.