On the (In)security of the Fiat-Shamir Paradigm

Shafi Goldwasser[*] Yael Tauman[†]

February 2, 2004

Abstract

In 1986, Fiat and Shamir proposed a general method for transforming secure 3-round public-coin identification schemes into digital signature schemes. The idea of the transformation was to replace the random message of the verifier in the identification scheme, with the value of some deterministic"hash" function evaluated on various quantities in the protocol and on the message to be signed.

The Fiat-Shamir methodology for producing digital signature schemes quickly gained popularity as it yields efficient and easy to implement digital signature schemes. The most important question however remained open: are the digital signatures produced by the Fiat-Shamir methodology *secure*?

In this paper, we answer this question negatively. We show that there exist secure 3-round public-coin identification schemes for which the Fiat-Shamir transformation yields insecure digital signature schemes for *any* "hash" function used by the transformation. This is in contrast to the work of Pointcheval and Stern which proved that the Fiat-Shamir methodology always produces digital signatures secure against chosen message attack in the "Random Oracle Model" – when the hash function is modelled by a random oracle.

Among other things, we make new usage of Barak's technique for taking advantage of non black-box access to a program, this time in the context of digital signatures.

[*] Department of Computer Science and Applied Math, The Weizmann Institute of Science at Rehovot, Israel and The Department of Computer Science and Electrical Engineering at MIT. Email: `shafi@theory.lcs.mit.edu`

[†] Department of Computer Science and Applied Math, The Weizmann Institute of Science, Rehovot 76100, Israel.

Contents

1 Introduction **3**

1.1 Our Results . . . 5

1.2 Related Work: Fiat-Shamir Paradigm and Zero Knowledge . . . 6

2 Preliminaries **7**

2.1 Identification Schemes . . . 9

2.1.1 Security of ID Schemes . . . 10

2.2 Signature Schemes . . . 10

2.2.1 Security of Signature Schemes . . . 11

2.3 The Fiat-Shamir Paradigm . . . 11

3 Proof of Theorem 1 **12**

4 Central Relation $\mathcal{R}_{\mathcal{F}}$ **14**

5 Interactive Arguments for $\mathcal{R}_{\mathcal{F}}$ **17**

5.1 First Interactive Argument: (P^0, V^0) . . . 18

5.2 Modified Interactive Argument: (P^1, V^1) . . . 20

5.3 Reduced-Interaction Argument: $(P^{\mathcal{H}}, V^{\mathcal{H}})$ . . . 21

5.3.1 $(P^{\mathcal{H}}, V^{\mathcal{H}})$ and CS-Proofs . . . 23

6 Proof of Theorem 2 **24**

6.1 Construction of ID^1 . . . 24

6.1.1 On the Insecurity of $SIG^1_{\mathcal{G},\mathcal{H}}$ . . . 25

6.1.2 On the Security of ID^1 . . . 25

6.2 Construction of ID^2 . . . 26

6.2.1 The Security of ID^2 . . . 28

6.2.2 On the Insecurity of $SIG^2_{\mathcal{H}}$ . . . 29

6.3 Construction of ID^3 . . . 32

7 On the Failure of FS Modifications **35**

7.1 First Modification . . . 36

7.2 Second Modification . . . 36

8 Future Directions **37**

9 Acknowledgements **38**

A Commitment Schemes **40**

B Proof of Lemma 6.4 **42**

C Proof of Claim 6.5.1 **46**

D Proof of Claim 6.7.1 **47**

E Proof of Claim 6.7.2 **48**

1 Introduction

In their famous paper laying the foundations for modern cryptography, Diffie and Hellman [DH76] introduced the notion of *digital signatures* and proposed a general method for designing them. Their method uses trapdoor functions as its basic primitive and is known as the *trapdoor-function signature method.* Several drawbacks of the trapdoor function approach have surfaced. In terms of security, by its very definition, it is prone to *existential forgery* as defined in [GMR88]. In terms of efficiency, the time to sign and verify are proportional to the time to invert and compute the underlying trapdoor function – a cost, which for some trapdoor functions, is prohibitive for certain applications.

Addressing the security concerns inherent in the trapdoor function model several other digital signature schemes were proposed and proved existentially unforgeable against chosen message attacks under standard intractability assumptions [GMR88, BM84, NY89, GHR99, CS99]. Most notably, [NY89] and [Rom90] showed that the existence of secure digital signature schemes is equivalent to the existence of one-way functions. These schemes, however, are rarely used in applications as they are often considered too inefficient.

A general paradigm for designing digital signature schemes was proposed by Fiat and Shamir [FS86]. Their starting observation was that designing secure interactive identification protocols (in which a sender merely identifies himself to a receiver) can be done with greater ease and efficiency than seems to be the case for secure digital signature schemes (in which a signer produces digital signatures for messages to be verified valid by a verifier). Building on this observation, they proposed a two-step approach for how to design secure digital signatures.

- First, design a secure 3-round public-coin identification scheme. Namely, a secure 3-round identification scheme $(\alpha, ; \beta; \gamma)$ where α is the prover's first message, β is a random message sent by the verifier, and γ is the prover's response.

- Second, obtain a digital signature scheme as follows. Let the signer publish a "hash" function h as part of his public-key. To sign a message M, the legal signer produces an accepting transcript of the interactive identification protocol $(\alpha;\beta;\gamma)$, where $\beta = h(\alpha, M)$. The legal signer who knows the secret key can produce accepting transcripts for any M. The intuition for why this signature scheme is secure is that when h is a sufficiently complicated function chosen by the the real signer it should be hard for a forger to find any message M and a transcript $(\alpha;\beta;\gamma)$ for which it is true both that $\beta = h(\alpha, M)$ and that γ is an answer which makes $(\alpha;\beta;\gamma)$ an accepting transcript of the identification protocol.

The complexity of a digital signature scheme resulting from the above paradigm is equivalent to the complexity of the starting identification scheme and the cost of evaluating the public function h. Current proposals for a public (keyless) function h are very efficient [MD5].

Due to the efficiency and the ease of design, the Fiat-Shamir paradigm quickly gained much popularity both in theory and in practice. Several digital signature schemes, including [Sch91, GQ88, Ok92], were designed following this paradigm. The paradigm has also been applied in other domains so as to achieve forward secure digital signature schemes [AABN02] and to achieve better exact security [MR02]. Both of the above applications actually use a variation of the Fiat-Shamir paradigm. Still, they share the same basic structure: start with some secure 3-round identification scheme and transform it into a digital signature scheme, eliminating the random move of the verifier by an application of a fixed function h to different quantities determined by the protocol and to the message to be signed.

The main question regarding any of these proposals is what can be proven about the security of the resulting signature schemes.

Pointcheval and Stern [PS96] made a first step towards answering this question. They proved that for every 3-round public-coin identification protocol, which is zero-knowledge with respect to an honest verifier, the signature scheme, obtained by applying the Fiat-Shamir transformation, is secure in the *Random Oracle Model.* This work was extended by Abdalla et. al. [AABN02] to show necessary and sufficient conditions on 3-round identification protocols for which the signature scheme, obtained by applying the Fiat-Shamir transformation, is secure in the Random Oracle Model.[1]

The Random Oracle Model is an *idealization* which assumes that all parties (including the adversary) have oracle access to a truly random function. The so called *random oracle methodology* is a popular methodology that uses the Random Oracle Model for designing cryptographic schemes. It consists of two steps. First, design a secure scheme in the Random Oracle Model. Then, replace the random oracle with a function, chosen at random from some function ensemble, and provide all parties (including the adversary) with a succinct description

[1] The conditions are that the identification scheme is secure against impersonation under passive attacks, and that the first message sent by the prover is drawn at random from a large space. [AABN02] show that the latter can be removed for a randomized version of the Fiat-Shamir transformation.

of this function. This gives an *implementation* of the idealized scheme in the real world. This methodology, introduced implicitly by [FS86], was formalized by Bellare and Rogaway [BR93].

As attractive as the methodology is for obtaining security "proofs", the obvious question was whether it is indeed always possible to replace the random oracle with a real world implementation. This question was answered negatively by Canetti, Goldreich and Halevi [CGH98]. They showed that there exists a signature scheme and an encryption scheme which are secure in the Random Oracle Model but are insecure with respect to any implementation of the random oracle by a function ensemble, thus showing that the random oracle methodology fails "in principle."

The work of [CGH98] left open the possibility that for particular "natural" cryptographic practices, such as the Fiat-Shamir paradigm, the random oracle methodology does work.

In this paper we show that this is not the case.

1.1 Our Results

We prove that the Fiat-Shamir paradigm for designing digital signatures can lead to universally forgeable digital signatures. We do so by demonstrating the existence of a secure 3-round public-coin identification scheme for which the corresponding signature scheme, obtained by applying the Fiat-Shamir paradigm, is insecure with respect to any function ensemble implementing the public function.

Our result relies on the existence of one-way functions. Note, however, that if one-way functions do not exist then secure signature schemes do not exist and thus the Fiat-Shamir paradigm always fails to produce secure signature schemes, as none exist. In this sense, our result is unconditional. Moreover, the problems we demonstrate for the Fiat-Shamir paradigm apply to all other variations of the Fiat-Shamir paradigm proposed in the literature [MR02, AABN02].

We stress that our result does not imply that particular ID schemes such as [FS86, Sch91] cannot be proven to yield secure signature schemes, with respect to some tailor-made function $\mathcal{H}$, under the Fiat-Shamir paradigm. What it does imply is that any proof of security would have to involve the particulars of the ID scheme and the $\mathcal{H}$ in question.

Our first idea is to make use of Barak's technique [Bar01] of taking advantage of non black-box access to the program of the verifier. Intuitively, the idea is to take any secure 3-round public-coin identification scheme (which is not necessarily zero-knowledge) and extend its verdict function so that the verifier also accepts views which convince him that the prover knows the verifier's next message. Since the verifier chooses the next message at random, there is no way that the prover can guess the verifier's next message during a real interaction, except with negligible probability, and therefore the scheme remains secure. However, when the identification scheme is converted into a signature scheme, by applying the Fiat-Shamir paradigm, the “verifier’s next message” is computed by a public function which is chosen at random from some function ensemble and is known in advance to everyone. A forger, who will now know in advance the “verifier’s next message” on any input, will be able to generate an accepting view for the verifier. This makes the signature scheme insecure regardless of which function ensemble is used to compute the “verifier’s next message” in the identification scheme.

The main technical challenge with implementing this approach is the following: How can the prover convince the verifier that he knows the verifier’s next message using a 3-round protocol?

We make strong use of the non-interactive CS-proofs of Micali [Mi94] to overcome this challenge. However, non-interactive CS-proofs themselves are only known to hold in the Random Oracle Model, and thus we *first* get the (somewhat odd-looking) conditional result that if CS-proofs are realizable in the real world by some function ensemble, then there exists a secure identification scheme for which the Fiat-Shamir paradigm always fails in the real world for all hash-function ensembles. Next, we show that even if CS-proofs are not realizable in the real world by any function ensemble, then again the Fiat-Shamir paradigm fails. This part of the proof contains the bulk of difficulty and technical complication. It entails showing different extensions of secure 3-round public-coin identification schemes, which become insecure as digital signature schemes when the Fiat-Shamir paradigm is applied to them. All in all, we construct three ID schemes ID^1, ID^2 and ID^3, and prove that at least one of them demonstrates the failure of the Fiat-Shamir paradigm.

1.2 Related Work: Fiat-Shamir Paradigm and Zero Knowledge

Following the work of [CGH98], Dwork, Naor, Reingold and Stockmeyer [DNRS99] investigated the security of the Fiat-Shamir paradigm, and showed that it is closely related to two previously studied problems: *the selective decommitment problem*[2], and *the existence of* 3-*round public-coin weak zero-knowledge arguments for non BPP languages*. We note that our negative results, regarding the security of the Fiat-Shamir paradigm, have implications on these related problems.

In particular, the result of [DNRS99], that the existence of 3-round public-coin zero-knowledge protocols for non BPP languages implies the insecurity of the Fiat-Shamir paradigm, is worth elaborating on. It follows from the following simple observation. Suppose there exists a 3-round public-coin zero-knowledge argument for some hard language. View this

[2] In the selective decommitment problem, an adversary is given commitments to a collection of messages, and the adversary can ask for some subset of the commitments to be opened. The question is whether seeing the decommitments to these open plaintexts allows the adversary to learn something unexpected about the plaintexts that are still hidden.

zero-knowledge argument as a secure identification protocol.[3] The fact that the identification protocol is zero-knowledge (and not only honest verifier zero-knowledge) means that for *every verifier* there exists a simulator that can generate identical views to the ones produced during the run of the identification protocol. As the Fiat-Shamir paradigm (applied to this identification protocol) essentially fixes a public program for the verifier of the zero-knowledge argument, any forger can now simply run the simulator for this fixed verifier to produce a view of the identification protocol, i.e. a valid digital signature.

This simple argument extends to any k-round public-coin zero-knowledge argument. Namely, if such a k-round public-coin zero-knowledge argument exists, it can be viewed as an identification protocol. Now, extend the original Fiat-Shamir paradigm to an *Extended-Fiat-Shamir* paradigm which replaces each message of the verifier (one round at a time) by applying a fixed public function to previous messages in the protocol. Then the same argument as above says, that the simulator for the k-round zero-knowledge protocol can be used to produce forgeries in the signature scheme resulting from the Extended-Fiat-Shamir paradigm, and thus the Extended-Fiat-Shamir paradigm fails.

Barak [Bar01] has shown that under the assumption that collision resistant function ensembles exist, every language in NP has a k-round (for some constant $k > 3$) public-coin zero-knowledge argument. Thus, it follows from [DNRS99] and [Bar01] that the k-round Extended-Fiat-Shamir paradigm is insecure.

However, the Fiat-Shamir paradigm was defined, and has always been used, only for 3-round identification schemes. Barak's work does not apply to this case. Moreover, whereas all that can be deduced from [DNRS99, Bar01] is that the Fiat-Shamir paradigm (extended or otherwise) fails on zero-knowledge identification schemes (indeed it is the simulator for the zero-knowledge system which will produce forgeries), it leaves open the possibility that the (extended and ordinary) Fiat-Shamir paradigm works when the starting identification schemes are secure with respect to a less strict security requirement and are not zero-knowledge.

2 Preliminaries

Notations: We use [GMR88]'s notations and conventions for probabilistic algorithms.
If $\mathcal{A}$ is a probabilistic algorithm then for any input x we let $\mathcal{A}(x)$ refer to the probability space which assigns to any string σ the probability that $\mathcal{A}(x)$ outputs σ. If S is a probability space then $x \leftarrow S$ denotes the algorithm which assigns to x an element randomly selected according to S. For any probabilistic interactive Turing machines A and B, we let $(A, B)(x)$ refer to the transcript of their interaction on input x. At the end of the interaction B will always either

[3] It is not necessarily a proof of knowledge but it is certainly a proof of ability of proving membership in L, which is hard for polynomial-time impersonating algorithms.

accept or reject. We refer to this decision function of B as the verdict function of B. We abuse notion by saying that $(A, B)(x) = 1$ if B accepts. we denote by $VIEW(B(x))$ the set of all transcripts that $B(x)$ accepts. We denote by $A|_\alpha$, machine A, restricted to sending α as its first message. More generally, we denote by $A|_{\alpha_1,\dots,\alpha_t}$, machine A, restricted to sending α_i as its i'th message, for $i = 1, \dots, t$.

Definition 1. (Negligible): *We say that a function $g(\cdot)$ is negligible if for every polynomial $p(\cdot)$ there exists $n_0 \in \mathbb{N}$ such that for every $n \geq n_0$*

$$g(n) < \frac{1}{p(n)}.$$

For any function $g(\cdot)$, we let $g(n) = negl(n)$ denote that $g(\cdot)$ is a negligible function.

Definition 2. (Non-negligible): *We say that a function $g(\cdot)$ is non-negligible if it is not negligible. That is, we say that $g(\cdot)$ is non-negligible if there exists a polynomial $p(\cdot)$ such that for infinitely many n's*

$$g(n) \geq \frac{1}{p(n)}.$$

For any function $g(\cdot)$, we let $g(n) = non\text{-}negl(n)$ denote that $g(\cdot)$ is a non-negligible function.

Definition 3. (one-way function): *We say that a polynomial-time computable function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ is one-way if for every polynomial-size circuit $C = \{C_n\}_{n \in \mathbb{N}}$,*

$$Pr[C_{|y|}(y) = x \ s.t. \ f(x) = y] = negl(n)$$

(where the probability is over uniformly chosen $y \in f(U_n)$).

Definition 4. (collision resistant hash-function ensemble): *We say that a hash-function ensemble $\mathcal{F} = \{\mathcal{F}_n\}_{n \in \mathbb{N}}$ is collision resistant if for every polynomial-size circuit $C = \{C_n\}_{n \in \mathbb{N}}$,*

$$Pr[C_n(f_n) = (x_1, x_2) \ s.t. \ f_n(x_1) = f_n(x_2)] = negl(n)$$

(where the probability is over uniformly chosen $f_n \in \mathcal{F}_n$).[4]

Definition 5. (Commitment Scheme): *A commitment scheme is a function ensemble $COMMIT = \{COMMIT_n\}_{n \in \mathbb{N}}$, where $COMMIT_n = \{commit_k\}_{k \in KEY_n}$, and there exist functions $l(n)$ and $t(n)$, which are polynomially-related to n, such that for every $n \in \mathbb{N}$ and every $k \in KEY_n$, $commit_k : \{0,1\}^n \times \{0,1\}^{l(n)} \rightarrow \{0,1\}^{t(n)}$, and the following properties are satisfied.*

[4] Throughout this paper we identify the description of a function $f \in \mathcal{F}_n$ with the seed used to generate it.

- *(Computationally-hiding): For every $n \in \mathbb{N}$, given any $k \in KEY_n$ and any $x \in \{0,1\}^n$, $commit_k(x;r) \cong U_{t(n)}$, assuming $r \cong U_{l(n)}$ (where $\cong$ denotes computational-indistinguishability).*

- *(Computationally-binding): For every $n \in \mathbb{N}$, given a random key $k \in_R KEY_n$ it is hard to find $(x_1, r_1) \neq (x_2, r_2)$ such that $commit_k(x_1; r_1) = commit_k(x_2; r_2)$. That is, for every polynomial-size circuit $C = \{C_n\}_{n\in\mathbb{N}}$*

$$Pr[C_n(k) = ((x_1, r_1), (x_2, r_2)) : commit_k(x_1; r_1) = commit_k(x_2; r_2)] = negl(n)$$

  (where the probability is over a uniformly chosen $k \in_R KEY_n$).

It was proven by Naor in [Na91] that commitment schemes exist, assuming the existence of one-way functions.

For the purposes of this paper, we need a special commitment scheme, which we denote by $COMM = \{COMM_n\}_{n\in\mathbb{N}}$, with the property that for any polynomial $m(\cdot)$, for every $n \in \mathbb{N}$ and for every $k \in KEY_n$, $COMM_k : \{0,1\}^{m(n)} \times \{0,1\}^n \to \{0,1\}^n$.[5] In Appendix A we show that such a commitment scheme exists (for any polynomial $m(\cdot)$), assuming collision resistant hash-function ensembles exist.

Next we define the notions of identification schemes and signature schemes, using standard definitions (see [GMR88, FFS88, Gol01]).

2.1 Identification Schemes

Definition 6. (Identification Scheme): *An identification scheme (or ID scheme, for short) is identified with a triplet (G, S, R), where G is a key generation algorithm and S is the sender who wishes to prove his identity to the Receiver R. More formally,*

- *G is a probabilistic-polynomial-time Turing machine that, on input 1^n, outputs a pair (SK, PK), such that the sizes of SK and PK are polynomially related to n. (SK is referred to as the secret-key and PK is referred to as the public-key).*

- *(S, R) is a pair of probabilistic-polynomial-time interactive Turing machines that take a public-key PK as common input. The sender S also takes a corresponding secret-key SK. It is required that for any pair (SK, PK) in the range of $G(1^n)$,*

$$Pr[(S(SK), R)(PK) = 1] = 1$$

  (where the probability is over the random coin tosses of S and R).

[5] Note that $COMM$ has the property that the size of the randomness equals the size of the commitment. We need this property since in the sequel we use one commitment as randomness for another commitment.

In this paper we are interested in a special type of ID scheme, which we refer to as a canonical ID scheme.

Definition 7. (Canonical ID Scheme): *A canonical ID scheme is a* 3*-round ID scheme* $(\alpha;\beta;\gamma)$*, in which* α *is sent by the sender* S*,* β *is sent by the receiver* R *and consists of* R*'s random coins, and* γ *is sent by the sender* S*.*

For a sender S, with keys (SK, PK) and randomness r, we denote $\alpha = S_{(SK,PK)}(r)$ and $\gamma = S_{(SK,PK)}(\alpha, \beta; r)$.

2.1.1 Security of ID Schemes

As with any cryptographic primitive, the notion of security considers adversary goals (what it has to do to win) and adversary capability (what attacks it is allowed). Naturally, for an ID scheme, the adversary's goal is impersonation: it wins if it can interact with the receiver (in the role of a sender), and convince the latter to accept. There are two natural attacks to consider: passive and active. Passive attacks correspond to eavesdropping, meaning the adversary is in possession of transcripts of conversations between the real sender and the receiver. Active attacks means that it gets to play the role of a receiver, interacting with the real sender in an effort to extract information. We note that assuming the existence of one-way function ensembles, there exist ID schemes which are secure against active attacks.[6] Throughout this paper, security of an ID scheme should be interpreted as security against active attacks.

2.2 Signature Schemes

Definition 8. (Signature Scheme): *A signature scheme is identified with a triplet* $(GEN, SIGN, VERIFY)$ *of probabilistic-polynomial-time Turing machines, where*

- GEN*, is the key generation algorithm which takes as input a security parameter* 1^n *and outputs a pair* (SK, VK) *known as the signing-key and the verification-key. Without loss of generality, we assume that the sizes of* SK, VK *are polynomially related to* n*.*
- $SIGN$ *is the signing algorithm which takes as input a pair* (SK, VK) *and a message* M *to be signed, and outputs a signature of* M *with respect to* (SK, VK)*.*

[6] This is the case since the existence of one-way function ensembles imply the existence of secure signature schemes [NY89], which in turn imply the existence of ID schemes which are secure against active attacks (see Section **??**).

- *$VERIFY$ is the verification algorithm which takes as input a verification-key VK, a message M and a string c (supposedly a signature of M with respect to VK), and outputs* 0 *or* 1. *Intuitively, it outputs 1 if c is a valid signature of M with respect to VK and it outputs 0 otherwise.*

Formally, it is required that for any pair (SK, VK) in the range of $GEN(1^n)$ and for any message M,

$$Pr[VERIFY(VK, M, SIGN((SK, VK), M)) = 1] = 1$$

(where the probability is over the random coin tosses of $SIGN$ and $VERIFY$).

2.2.1 Security of Signature Schemes

Several types of security requirements were considered in the literature. We say that a signature scheme is secure if it is existentially secure against adaptive chosen message attacks.

Definition 9. *(*Security against adaptive chosen message attacks): *We say that a signature scheme $SIG = (GEN, SIGN, VERIFY)$ is secure if for every polynomial-size circuit family $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, with oracle access to $SIGN$, the probability that, on input a uniformly chosen verification-key $VK \leftarrow GEN(1^n)$, $\mathcal{F}_n$ outputs a pair (M_0, SIG_{M_0}) such that $VERIFY(VK, M_0, SIG_{M_0}) = 1$ and such that M_0 was not sent by $\mathcal{F}_n$ as an oracle query to $SIGN$, is negligible (where the probability is over VK and over the randomness of the oracle SIGN).*

2.3 The Fiat-Shamir Paradigm

Definition 10. (The Fiat-Shamir Paradigm): *Given any canonical ID scheme $ID = (G, S, R)$ and any hash-function ensemble $\mathcal{H} = \{\mathcal{H}_n\}_{n\in\mathbb{N}}$, the Fiat-Shamir paradigm transforms ID and $\mathcal{H}$ into a signature scheme $SIG_{\mathcal{H}} = (GEN_{\mathcal{H}}, SIGN_{\mathcal{H}}, VERIFY_{\mathcal{H}})$, defined as follows.*

- *The key generation algorithm $GEN_{\mathcal{H}}$, on input 1^n emulates algorithm $G(1^n)$ to generate a pair (SK, PK) of secret key and public key. It then chooses at random a function $h \in \mathcal{H}_n$, and outputs SK as the signing key and $VK = (PK, h)$ as the verification key.*

- *The signing algorithm $SIGN_{\mathcal{H}}$, on input a signing key SK, a corresponding verification key $VK = (PK, h)$, and a message M, emulates the sender S with respect to (SK, PK) to produce (α, β, γ), where $\beta = (\alpha, M)$. That is, $SIGN_{\mathcal{H}}(SK, VK, M)$ operates as follows.*

    1. *Tosses coins r (for S) and computes $\alpha = S_{(SK,PK)}(r)$.*

2. *Computes $\beta = h(\alpha, M)$.*
3. *Computes $\gamma = S_{(SK,PK)}(\alpha, \beta; r)$.*
4. *Outputs (α, β, γ) as a signature of M.*

- *The verification algorithm $VERIFY_{\mathcal{H}}$, on input a verification-key $VK = (PK, h)$, a message M and a triplet (α, β, γ) (which is supposedly a signature of M), accepts if and only if $\beta = h(\alpha, M)$ and $(\alpha; \beta; \gamma) \in VIEW(R(PK))$.*

Throughout this paper, the Fiat-Shamir paradigm is referred to as the FS paradigm. We denote by $FS_{\mathcal{H}}(ID)$ the signature scheme obtained by applying the FS paradigm to ID and $\mathcal{H}$.

We say that the the FS paradigm is *secure* if for every secure canonical ID scheme ID, there exists a hash-function ensemble $\mathcal{H}$ such that $FS_{\mathcal{H}}(ID)$ is secure. Otherwise, we say that the FS paradigm *fails*. We denote by (FS) the case that the FS paradigm is secure and we denote by $\neg(FS)$ the case that the FS paradigm fails.

We note that the FS paradigm, taken outside of the context of ID schemes and digital signature schemes, provides a general way of eliminating interaction from protocols by replacing the verifier with a function ensemble. As such, it has also been applied in other contexts, such as in the context of CS proofs [Mi94].

In the rest of the paper we focus on proving the following two theorems.

Theorem 1. *If collision resistant hash-function ensembles do not exist and one-way functions do exist then the FS paradigm fails.*

Theorem 2. *If collision resistant hash-function ensembles exist then the FS paradigm fails.*

Corollary 3. *If one-way functions exist then the FS paradigm fails.*

It is well known that if one-way functions do not exist then neither do secure digital signature schemes. Thus, in a sense our result is unconditional since we get that the FS paradigm either fails or is useless (i.e., never produces secure digital signatures, as none exist).

We note that the proof of the first theorem is relatively simple and that the main contribution of this paper is in proving the second theorem. In what follows we give the main ideas in the proofs of the above two theorems.

3 Proof of Theorem 1

In this subsection, we assume that collision resistant hash-function ensembles do not exist and that one-way functions do exist. That is, we assume that for every hash-function ensemble

$\mathcal{H} = \{\mathcal{H}_n\}$, for infinitely many n's, given a random $h \in \mathcal{H}_n$, it is easy to find $m_1 \neq m_2$ such that $h(m_1) = h(m_2)$. For every $\mathcal{H}$, we denote the set of all such n's by $S_{\mathcal{H}}$. Our goal is to construct a secure canonical ID scheme ID such that for every $\mathcal{H}$, the corresponding signature scheme $FS_{\mathcal{H}}(ID)$ will be insecure. More specifically, we will demonstrate the insecurity of $FS_{\mathcal{H}}(ID)$ by constructing a forger that for every $n \in S_{\mathcal{H}}$ will succeed in forging signatures, with respect to $VK = (PK, h)$ generated by $GEN(1^n)$, with non-negligible probability.

Intuitively, ID will be defined as follows. Fix any secure signature scheme $SIG = (GEN, SIGN, VERIFY)$ (the existence of secure signature schemes follows from the existence of one-way functions [Rom90, NY89]). The sender will identify himself by signing a random message sent by the receiver.[7] The security of ID will follow from the security of SIG. The insecurity of $FS_{\mathcal{H}}(ID)$ will follow from the assumption that collision resistant hash-function ensembles do not exist.

Proof. Let $SIG = (GEN, SIGN, VERIFY)$ be any secure signature scheme. Consider the following ID scheme, $ID = (G, S, R)$.

- G: On input 1^n, emulate $GEN(1^n)$ to obtain a pair (SK, VK), and output SK as the secret-key and VK as the public-key.

- S and R are interactive Turing machines, that for any $(SK, VK) \leftarrow G(1^n)$, the interaction of $(S(SK), R)(VK)$ is as follows.

$$\begin{array}{ccc} S(SK) & VK & R \\ & \xrightarrow[\emptyset]{\qquad\qquad} & \\ & \xleftarrow[\beta]{\qquad\qquad} & \\ & \xrightarrow[SIGN((SK,VK)(\beta))]{} & \end{array}$$

  $R(VK)$ accepts a transcript $(\alpha; \beta; \gamma)$ if and only if $\alpha = \emptyset$ and $VERIFY(VK, \beta, \gamma) = 1$ (i.e., γ is a valid signature of β, with respect to the verification-key VK).

Claim 3.0.1. *ID is secure, assuming SIG is a secure signature scheme.*

Proof. Trivial! □

[7] Note that in some sense this is the inversion of the Fiat-Shamir paradigm, which converts any secure canonical ID scheme into a signature scheme.

Claim 3.0.2. *$FS_{\mathcal{H}}(ID) = (GEN_{\mathcal{H}}, SIGN_{\mathcal{H}}, VERIFY_{\mathcal{H}})$ is insecure assuming collision resistant hash-function ensembles do not exist.*

Proof. A forger, given a verification-key $(VK, h) \leftarrow GEN(1^n)$, where $n \in S_{\mathcal{H}}$, and given a signing oracle, will forge a signature to some new message M, as follows.

1. Find $M_1 \neq M_2$ such that $h(M_1) = h(M_2)$. From our assumption this can be done by a poly-size circuit.

2. Query the signing oracle with the message M_1, to obtain a signature (α, β, γ).

3. Output (α, β, γ) as a signature to M_2.

Notice that (α, β, γ) is a valid signature of M_2 if it is a valid signature of M_1 and $h(M_1) = h(M_2)$. Since both of these conditions are satisfied with non-negligible probability, the forger succeeds in forging a signature of M_2 with non-negligible probability. □

□

Throughout the rest of the paper we assume the existence of a collision resistant hash-function ensemble, which we denote by $\mathcal{F}$. Actually, we restrict our attention to collision resistant hash-function ensembles from $\{0,1\}^{2n}$ to $\{0,1\}^n$.

4 Central Relation $\mathcal{R}_{\mathcal{F}}$

Recall that our goal is to construct a secure canonical ID scheme ID such that for any hash-function ensemble $\mathcal{H}$, $FS_{\mathcal{H}}(ID)$ will be an insecure digital signature scheme. Our first idea towards achieving this goal if the following.

Take any secure canonical ID scheme and extend its verdict function so as to also accept transcripts which convince the receiver that the sender knows in advance the receiver's next message. Since the receiver chooses the next message at random (by definition of a canonical ID scheme), there is no way that a sender can guess in advance the receiver's next message, except with negligible probability, and therefore the scheme remains secure. However, when the ID scheme is converted into a signature scheme via the FS paradigm, the receiver is replaced with a succinct public function, and thus everyone knows in advance the "receiver's next message" on any input, and so can generate an accepting transcript, which corresponds to a legitimate signature. Hence, the corresponding signature scheme will be insecure with respect to any hash-function ensemble.

The main problem with this approach is the following: How can the sender convince the receiver that he knows the receiver's next message? One idea is for the sender to send the receiver a polynomial-size circuit which computes the receiver's next message. The problem is that we must first fix the ID scheme (in particular, fix a polynomial bound on the size of its messages) and only then show that for *any* hash-function ensemble $\mathcal{H}$ replacing the receiver in the signature scheme, $FS_{\mathcal{H}}(ID)$ is insecure. In other words, we need to find a protocol of a-priori bounded size, in which the sender will be able to convince the receiver of knowledge of *any* polynomial-size circuit corresponding to *any* $\mathcal{H}$.

To achieve this goal, the sender, instead of sending his circuit in hand (which may be too big), will send a size-reducing commitment to his circuit. The type of commitment we use is a tree-commitment, which allows a fixed polynomial-size commitment for any polynomial-size string. The notion of *tree-commitment* was introduced by Merkle [Mer90].

Definition 11. (Tree-Commitment): *A tree-commitment to $x \in \{0,1\}^*$, with respect to the function $f : \{0,1\}^{2n} \to \{0,1\}^n$, is defined as follows. Consider a complete binary tree of depth $\lg(|x|/n)$, where each node has a label in $\{0,1\}^n$. The leaves are labeled by the bits of x (n bits per leaf). Each internal node is labeled by applying f to the label of its children. The tree-commitment to x, with respect to f, is denoted by $TC_f(x)$, and consists of the label of the root and the depth of the tree.*[8]

Note that a tree-commitment is not only useful for its size-reducing property, but it also has the advantageous property that it allows decommitment to individual bits. We let $auth_f(x,i)$ denote the authentication path of the ith bit of x with respect to f. Namely, $auth_f(x,i)$ consists of the label of x_i's siblings, the labels of its ancestors and the labels of its ancestors siblings. We let $auth_f(x)$ denote the entire tree, which contains the authentication path of x_i, for every i.

We are now ready to define the ID scheme which will supposedly prove the failure of the FS paradigm. Start with any secure ID scheme ID and extend its verdict function so as to also accept views in which the sender first sends message a (supposedly a tree-commitment to a circuit C), the receiver replies with b, and only then the sender proves to the receiver that he knows a circuit C, such that both $TC_f(C) = a$ and $C(a) = b$ (where f is pre-specified and chosen at random from a collision resistant hash-function ensemble $\mathcal{F}$). More precisely, the sender proves that he knows a circuit C, which is a witness to (f, a, b) in the following relation:

Definition 12. (Central Relation):

$$\mathcal{R}_{\mathcal{F}} = \{((f,a,b), auth_f(\hat{C})) \ : \ TC_f(\hat{C}) = a \ \wedge \ C(a) = b \ \wedge \ |\hat{C}| < n^{\lg n}\}$$

where $C \to \hat{C}$ is a special circuit-encoding which satisfies the following properties.

[8] Note that if f is chosen at random from a collision resistant hash-function ensemble then the tree-commitment, with respect to f, is computationally binding.

1. *It is an efficient encoding. Namely, there is a polynomial-time algorithm that given any circuit C, outputs $\hat{C}$.*

2. *It has high minimum distance. Namely, for every $C_1 \neq C_2$, $\hat{C}_1$ and $\hat{C}_2$ differ in a polynomial fraction of their coordinates.*

3. *Given y, it is easy to check whether y is a codeword. Namely, there is a polynomial-time algorithm that given y, outputs 1 if and only if there exists a circuit C such that $y = \hat{C}$.*

4. *There exists a polynomial-time algorithm that given any circuit-encoding $\hat{C}$ (where C is defined on inputs of size n) and given any $x \in \{0,1\}^n$, computes $C(x)$.*

Remarks:

1. The reason we bound the size of $\hat{C}$ by $n^{\lg n}$ is because the receiver's 'next message' function can be of *any* polynomial-size. Hence, we cannot bound the size of $\hat{C}$ by a fixed polynomial, and so we bound it by some super-polynomial, such as $n^{\lg n}$.

2. We defined $\mathcal{R}_{\mathcal{F}}$ using a tree-commitment, as opposed to a regular length-reducing commitment, for the following technical reason. In our proof we get a contradiction to the security of the Fiat-Shamir paradigm, by claiming knowledge of $\hat{C}_1 \neq \hat{C}_2$ which commit to the same value. However, the size of these circuits is not a-priori bounded by some polynomial, and hence we will not be able to extract this knowledge using a polynomial-time algorithm. We get around this technical problem by using a tree-commitment, which allows one to decommit to individual bits.

3. Without loss of generality we assume that $auth_f(\hat{C})$ is of the following form: After every bit of $\hat{C}$ there are exactly $(\lg n)^2$ bits of the authentication path of that bit. Namely, we assume that the i'th bit of $\hat{C}$ is represented in the $(1 + (i-1)((\lg n)^2 + 1))$'th bit of $auth_f(\hat{C})$, followed by $(\lg n)^2$ bits of its authentication path. We will need this precision of representation in proving Lemma B.0.3[9].

4. Sometimes we refer to a witness of (f, a, b) by $w_{(f,a,b)}$.

Recall that in the above extended ID scheme, in the third round the sender needs to prove knowledge of a witness of (f, a, b), where a is the message sent by the sender in the first round, b is the message sent by the receiver in the second round, and f is a pre-specified collision resistant hash-function. Thus, in particular, we need one round proof-of-knowledge system

[9] Actually, the only reason we included the authentication path in the witness is to allow us to prove Lemma B.0.3.

for $\mathcal{R}_{\mathcal{F}}$. Note that actually, we need a proof-of-knowledge system for $\mathcal{R}_{\mathcal{F}}$ which is either one round, or two rounds in which the first round consists of the verifier's random coin tosses. Note that this is not an easy task as $\mathcal{R}_{\mathcal{F}}$ is not an NP-relation.

If there somehow existed a 2-round public-coin proof-of-knowledge system for $\mathcal{R}_{\mathcal{F}}$ then we would be done, since we could take the secure canonical ID scheme ID, extend its public-key by appending a random $f \in_R \mathcal{F}$ to it, and extend its verdict function so as to also accept transcripts of the form

$$\xrightarrow[a]{}$$

$$\xleftarrow[b,q]{}$$

$$\xrightarrow[ans]{}$$

where $(q; ans)$ is a 2-round public-coin proof-of-knowledge of C such that $((f,a,b),C) \in \mathcal{R}_{\mathcal{F}}$.

Unfortunately, we do not know whether a 2-round proof-of-knowledge system for $\mathcal{R}_{\mathcal{F}}$ exists.

5 Interactive Arguments for $\mathcal{R}_{\mathcal{F}}$

In this section we try to find a 2-round proof-of-knowledge for $\mathcal{R}_{\mathcal{F}}$.

Proposition 1. *[BG01]: $L_{\mathcal{R}_{\mathcal{F}}} \in NTIME(n^{\lg n})$.*

Proof. Follows immediately from the definition of $\mathcal{R}_{\mathcal{F}}$ and from properties 3 and 4 of the circuit-encoding $C \to \hat{C}$. □

From the theory on Probabilistic-Checkable-Proofs it follows that there exists a polynomial-time Turing machine P_{PCP} and a probabilistic-polynomial-time oracle machine V_{PCP} with the following properties.

1. (Relatively-efficient oracle construction): for every $((f,a,b), w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$, $P_{PCP}((f,a,b), w_{(f,a,b)}) = \pi$ such that $Pr[V^{\pi}_{PCP}(f,a,b) = 1] = 1$. Throughout the paper, we refer to π as a PCP proof.

2. (Non-adaptive verifier:) The verifier's queries are determined based only on its input and on its internal coin tosses. That is, there exists a probabilistic-polynomial-time algorithm Q_{PCP} such that on input (f,a,b) and random coins r, the verifier makes the query sequence $\{q_i\}$, where for every i, $q_i = Q_{PCP}((f,a,b),r,i)$.

3. (Efficient reverse-sampling): There exists a probabilistic-polynomial-time oracle machine S such that, on input any string (f,a,b) and integers i and q, outputs a uniformly distributed r that satisfies $Q_{PCP}((f,a,b),r,i) = q$.

4. (Proof-of-knowledge): There exists a probabilistic-polynomial-time oracle machine E and a negligible function $\epsilon(\cdot)$ such that, for every (f, a, b) and for every π, if $Pr[V^{\pi}_{PCP}(f, a, b) = 1] > \epsilon(|(f, a, b)|)$, then there exists w such that $((f, a, b), w) \in \mathcal{R}_{\mathcal{F}}$ and for every i, $Pr[E^{\pi}((f, a, b), i) = w_i] \geq 2/3$.

Based on the above theory of PCP, Barak and Goldreich [BG01], based on work of [Ki92, Mi94], presented a 4-round public-coin argument for every language in $NEXP$, and in particular for $\mathcal{R}_{\mathcal{F}}$. We begin by presenting this 4-round argument for $\mathcal{R}_{\mathcal{F}}$. We then do a series of modifications and obtain a reduced interaction version of this construction. We reduce interaction by applying the Fiat-Shamir paradigm itself, this time in the context of Universal Arguments. This seems like a strange idea, since our goal is to prove the failure of the FS paradigm, but it will take us one step further in the proof.

5.1 First Interactive Argument: (P^0, V^0)

- Common input: (f, a, b) (where $f \in \mathcal{F}_n$ and $a, b \in \{0, 1\}^n$).
- Auxiliary input to the prover: $w_{(f,a,b)}$ such that supposedly $((f, a, b), w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$.

1. V^0: Uniformly select $f^{UA} \in_R \mathcal{F}_n$ and send it to the prover.
2. P^0:
   (a) Construct a PCP proof of $((f, a, b), w_{(f,a,b)})$ by computing $\pi = P_{PCP}((f, a, b), w_{(f,a,b)})$.
   (b) Compute $\beta = TC_{f^{UA}}(\pi)$[10], which is the tree-commitment to π with respect to f^{UA}.
   (c) Send β to the prover.
3. V^0: Uniformly select a random-tape γ for V_{PCP}, and send γ to the prover.
4. P^0: Provide the answers to the (PCP) queries of $V_{PCP}((f, a, b); \gamma)$ augmented by proofs of consistency to these answers.
   (a) Determining the queries: Invoke $Q_{PCP}((f, a, b); \gamma)$, in order to determine the sequence of queries that V_{PCP} makes on input (f, a, b), given a random string γ.

[10] Note that there are two levels of use of the tree-commitment.

- In the definition of $\mathcal{R}_{\mathcal{F}}$: $TC_f(w_{(f,a,b)}) = a$.
- In the interactive argument for $\mathcal{R}_{\mathcal{F}}$: $TC_{f^{UA}}(\pi) = \beta$.

In both cases we use a tree-commitment since the size of both $w_{(f,a,b)}$ and π may be to large to extract. Using a tree-commitment we can extract only a few coordinates, with the ability to verify that these values were committed to.

(b) For every query q_i of $Q_{PCP}((f,a,b);\gamma)$, send the label of the leaf that contains π_{q_i} and send the labels of the path corresponding to this leaf, which consists of the label of its sibling, the labels of its ancestors and the labels of its ancestors siblings, which are needed in order to verify consistency with β.

We denote this response by $\delta = (label(\gamma), auth(\gamma))$.

V^0 accepts if and only if the answers provided by the prover would have been accepted by V_{PCP}, and all the proofs of consistency are valid.

(P^0, V^0), on input (f,a,b), can be schematically viewed as follows.

$$\begin{array}{ccc}
P^0(\hat{C}) & & V^0 \\
 & \xleftarrow[f^{UA}\in\mathcal{F}_n]{\qquad\qquad} & \\
\begin{array}{c}\pi = P_{PCP}((f,a,b), w_{(f,a,b)}) \\ \beta = TC_{f^{UA}}(\pi)\end{array} & \xrightarrow[\beta]{\qquad\qquad} & \\
 & \xleftarrow[\gamma]{\qquad\qquad} & \\
 & \xrightarrow[\delta = (label(\gamma), auth(\gamma))]{} &
\end{array}$$

Lemma 5.1. *[Mi94],[BG01]: (P^0, V^0) satisfies the following properties.*

- *(Completeness): For every $((f,a,b), w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$, $Pr[(P^0(w_{(f,a,b)}), V^0)(f,a,b) = 1] = 1$ (where the probability is over the random coin tosses of V^0).*
- *(CS-proof-of-knowledge): For every polynomial $p(\cdot)$, there exists a polynomial $p'(\cdot)$ and a probabilistic-polynomial-time oracle machine E such that for every polynomial-size circuit family $P^* = \{P^*_n\}$, for every sufficiently large n, and for every input (f,a,b), if $Pr[(P^*_n, V^0)(f,a,b) = 1] \geq 1/p(n)$ (where the probability is over the random coin tosses of V^0), then $Pr[\exists w$ s.t. $((f,a,b),w) \in \mathcal{R}_{\mathcal{F}}$ and $\forall i$ $E^{P^*_n}((f,a,b),i) = w_i] \geq 1/p'(n)$ (where the probability is over the random coin tosses of E).*

We will not prove this Lemma since it was proved in [BG01] (using the four properties of (P_{PCP}, V_{PCP})). Moreover, following the proof in [BG01], it can be easily seen that the above proof-of-knowledge property holds even if P^*_n chooses (f,a,b) after receiving the verifier's first message f^{UA}.

5.2 Modified Interactive Argument: (P^1, V^1)

For reasons to be clarified later, we modify slightly the above interactive argument, by modifying the prover's first message from β to a commitment of β. Formally, we define a modified interactive argument, which we denote by (P^1, V^1), as follows.

- Common input: (f, a, b) (where $f \in \mathcal{F}_n$ and $a, b \in \{0,1\}^n$).
- Auxiliary input to the prover: $w_{(f,a,b)}$ such that supposedly $((f,a,b), w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$.

1. V^1: Uniformly select
   - $f^{UA} \in \mathcal{F}_n$ (a function for the tree-commitment)
   - $k \in KEY_n$ (a seed for $COMM$)
   - $r \in \{0,1\}^n$ (randomness for $COMM$)

   Send $(f^{UA}, (k, r))$ to the prover.
2. P^1:
   (a) Construct a PCP-proof of $((f,a,b), w_{(f,a,b)})$ by computing $\pi = P_{PCP}((f,a,b), w_{(f,a,b)})$.
   (b) Compute $\beta = TC_{f^{UA}}(\pi)$ which is a tree-commitment to π with respect to f^{UA}.
   (c) Send $\hat{\beta} = comm_k(\beta; r)$.
3. V^1: Uniformly select a random-tape γ for V_{PCP}, and send γ to the prover.
4. P^1: Send β, along with $\delta = (label(\gamma), auth(\gamma))$, which consists of the answers to the (PCP) queries of $V_{PCP}((f,a,b); \gamma)$ augmented by proofs of consistency to these answers.

V^1 accepts if and only if $\hat{\beta} = comm_k(\beta; r)$ and $(f^{UA}, \beta, \gamma, \delta) \in VIEW(V^0(f,a,b))$.

(P^1, V^1), on input (f, a, b), can be schematically viewed as follows.

$$\begin{array}{ccc}
P^1(\hat{C}) & & V^1 \\
 & \xleftarrow[f^{UA}, (k, r)]{\hspace{6em}} & \\
\begin{array}{c} \pi = P_{PCP}((f, a, b), w_{(f,a,b)}) \\ \beta = TC_{f^{UA}}(\pi) \end{array} & \xrightarrow[\hat{\beta} = comm_k(\beta; r))]{\hspace{6em}} & \\
 & \xleftarrow[\gamma]{\hspace{5em}} & \\
\delta = (label(\gamma), auth(\gamma)) & \xrightarrow[\beta, \delta]{\hspace{6em}} &
\end{array}$$

Lemma 5.2. *(P^1, V^1) satisfies the following properties.*

- *(Completeness): For every $((f, a, b), w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$, $Pr[(P^1(w_{(f,a,b)}), V^1)(f, a, b) = 1] = 1$ (where the probability is over the random coin tosses of V^1).*
- *(CS-proof-of-knowledge): For every polynomial $p(\cdot)$, there exists a polynomial $p'(\cdot)$ and a probabilistic-polynomial-time oracle machine E such that for every polynomial-size circuit family $P^* = \{P^*_n\}$, for every sufficiently large n, and for every input (f, a, b), if $Pr[(P^*_n, V^1)(f, a, b) = 1] \geq 1/p(n)$ (where the probability is over the random coin tosses of V^1), then $Pr[\exists w \text{ s.t. } ((f, a, b), w) \in \mathcal{R}_{\mathcal{F}} \text{ and } \forall i\ E^{P^*_n}((f, a, b), i) = w_i] \geq 1/p'(n)$ (where the probability is over the random coin tosses of E).*

As before, the above proof-of-knowledge property holds even if P^*_n chooses (f, a, b) after receiving the verifier's first message $(f^{UA}, (k, r))$.

5.3 Reduced-Interaction Argument: $(P^{\mathcal{H}}, V^{\mathcal{H}})$

Next, we reduce the number of rounds by applying the Fiat-Shamir paradigm itself to (P^1, V^1) (i.e., by replacing V^1's second message with some function applied to P^1's first message).

For any function ensemble $\mathcal{H}$, we define a reduced-interaction argument $(P^{\mathcal{H}}, V^{\mathcal{H}})$ for $\mathcal{R}_{\mathcal{F}}$, with respect to $\mathcal{H}$, as follows.

- Common input: (f, a, b).

- Auxiliary input to the prover: $w_{(f,a,b)}$ such that supposedly $((f,a,b),w_{(f,a,b)}) \in \mathcal{R}_{\mathcal{F}}$.

1. $V^{\mathcal{H}}$: Uniformly select
   - $f^{UA} \in \mathcal{F}_n$ (a function for the tree-commitment)
   - $k \in KEY_n$ (a seed for $COMM$)
   - $r \in \{0,1\}^n$ (randomness for $COMM$)
   - $h_1,\ldots,h_n \in \mathcal{H}_n$

   Send $(f^{UA},(k,r),(h_1,\ldots,h_n))$ to the prover.

2. $P^{\mathcal{H}}$: For $i = 1,\ldots,n$,
   (a) Invoke P_{PCP} on $((f,a,b),w_{(f,a,b)})$ to obtain $\pi_i = P_{PCP}((f,a,b),w_{(f,a,b)})$.
   (b) Compute $\beta_i = TC_{f^{UA}}(\pi_i)$.
   (c) Compute $\hat{\beta}_i = comm_k(\beta_i;r)$.
   (d) compute $\gamma_i = h_i(\hat{\beta}_i)$.
   (e) Let δ_i be the (PCP) answers corresponding to the queries $Q_{PCP}((f,a,b);\gamma_i)$ augmented by proofs of consistency to these answers.

   send $\{\beta_i,\hat{\beta}_i,\gamma_i,\delta_i\}_{i=1}^n$.

$V^{\mathcal{H}}$ accept if and only if for $i = 1,\ldots,n$ the following conditions hold.

1. $\hat{\beta}_i = comm_k(\beta_i;r)$.
2. $\gamma_i = h_i(\hat{\beta}_i)$.
3. $(f^{UA},\beta_i,\gamma_i,\delta_i) \in VIEW(V^0(f,a,b))$.

$(P^{\mathcal{H}},V^{\mathcal{H}})$, on input (f,a,b), can be schematically viewed as follows.

$$
\begin{array}{ccc}
P^{\mathcal{H}}(\hat{C}) & & V^{\mathcal{H}} \\
 & \xleftarrow[f^{UA},(k,r),(h_1,\ldots,h_n)]{\hspace{6cm}} & \\
\begin{array}{c}
\pi_i = P_{PCP}((f,a,b),w_{(f,a,b)}) \\
\beta_i = TC_{f^{UA}}(\pi_i) \\
\hat{\beta}_i = comm_k(\beta_i;r) \\
\gamma_i = h_i(\hat{\beta}_i) \\
\delta_i = (label(\gamma_i),auth(\gamma_i))
\end{array} & \xrightarrow[\{\beta_i,\hat{\beta}_i,\gamma_i,\delta_i\}_{i=1}^n]{\hspace{6cm}} &
\end{array}
$$

Remarks on $(P^{\mathcal{H}}, V^{\mathcal{H}})$:

1. The reason that we require the prover to convince the verifier with n functions (rather than just one function) is to achieve error reduction.

2. We introduce some notation which will be useful later. Let q denote the message sent by $V^{\mathcal{H}}$, and let ans denote the response to q sent by $P^{\mathcal{H}}$. Recall that if $V^{\mathcal{H}}(f, a, b)$ accepts the view $(q; ans)$, then we say that $(q; ans) \in VIEW(V^{\mathcal{H}}(f, a, b))$.

It is easy to see that for every function ensemble $\mathcal{H}$, $(P^{\mathcal{H}}, V^{\mathcal{H}})$ satisfies the completeness requirement. However, we do not know if $(P^{\mathcal{H}}, V^{\mathcal{H}})$ satisfies the CS-proof-of-knowledge property.

5.3.1 $(P^{\mathcal{H}}, V^{\mathcal{H}})$ and CS-Proofs

The proof system $(P^{\mathcal{H}}, V^{\mathcal{H}})$ is closely related to CS-proofs, defined by Micali [Mi94], since CS-proofs are essentially a non-interactive version of (P^0, V^0) obtained by replacing the verifier V^0 with a random oracle. Micali proved that, in the Random Oracle Model, CS proofs satisfy both the completeness property and the CS-proof-of-knowledge property.[11] One can make the following hypothesis.

Hypothesis (CSP): *There exists a function ensemble $\mathcal{H}$ such that if the random oracle is replaced with a function uniformly chosen from $\mathcal{H}$, then CS-proofs still satisfy both the completeness property and the CS-proof-of-knowledge property.*

Looking carefully into the definition of CS-Proofs one can easily verify the following.

Proposition 2. *The CSP hypothesis implies that there exists a function ensemble $\mathcal{H}$ for which $(P^{\mathcal{H}}, V^{\mathcal{H}})$ satisfies both the completeness property and the CS-proof-of-knowledge property.*

This is quite surprising, since it essentially implies that if CS proofs exist in the real world, then the FS paradigm fails. Or in other words, if the FS paradigm applied to (P^0, V^0) results with a secure scheme, then the FS paradigm applied to canonical ID-schemes results with insecure schemes.

It turns out that the bulk of complication is in showing that if the CSP hypothesis is false then still the FS paradigm fails. In other words, the bulk of complication is in proving that if the FS paradigm, applied to (P^0, V^0), results with an insecure scheme, then the FS paradigm, applied to canonical ID-schemes, also results with insecure schemes. This is also surprising since we expected this direction to be the easy one.

[11] The definitions of completeness and of CS-proof-of-knowledge were given in Lemma 5.1 and Lemma 5.2.

6 Proof of Theorem 2

Our goal is to construct a secure canonical ID scheme ID such that for any hash-function ensemble $\mathcal{H}$, $FS_{\mathcal{H}}(ID)$ will be an insecure digital signature scheme. In fact we cannot point to one explicit construction of such an ID scheme. Instead, we show three explicit constructions of ID schemes: ID^1, ID^2, ID^3, and prove that the FS paradigm must fail with respect to one of the three.

6.1 Construction of ID^1

Let $\mathcal{F}$ be a collision resistant hash-function ensemble, let $\mathcal{G}$ be some a-priori fixed function ensemble, and let $ID = (G, S, R)$ be any secure canonical ID scheme. We extend ID to obtain a new ID scheme $ID^1_{\mathcal{G}} = (G^1, S^1, R^1)$, by extending the public-key and the verdict function of ID, as follows.

- G^1: on input 1^n,

  1. Run $G(1^n)$, to obtain a pair $(SK, PK) \leftarrow G(1^n)$.
  2. Choose $f \in_R \mathcal{F}_n$.

  Output SK as the secret-key and $PK' = (PK, f)$ as the public-key.

- R^1: On input a public-key $PK' = (PK, f)$, R^1 will accept either views that $R(PK)$ accepts or views of the form

$$\begin{array}{ccc} S^1 & & R^1 \\ & \xrightarrow[a]{\qquad\qquad} & \\ & \xleftarrow[b,q]{\qquad\qquad} & \\ & \xrightarrow[ans]{\qquad\qquad} & \end{array}$$

  such that $(q; ans) \in VIEW(V^{\mathcal{G}}(f, a, b))$.

To establish $\neg(FS)$, we need to show that the ID scheme $ID^1_{\mathcal{G}}$ is secure and that the signature scheme $FS_{\mathcal{H}}(ID^1_{\mathcal{G}})$ is insecure with respect to any function ensemble $\mathcal{H}$. We begin by proving the insecurity of $FS_{\mathcal{H}}(ID^1_{\mathcal{G}})$.

We denote $FS_{\mathcal{H}}(ID^1_{\mathcal{G}})$ by $SIG^1_{\mathcal{G},\mathcal{H}} = (GEN^1_{\mathcal{H}}, SIGN^1_{\mathcal{H}}, VERIFY^1_{\mathcal{H}})$.

6.1.1 On the Insecurity of $SIG^1_{\mathcal{G},\mathcal{H}}$

Lemma 6.1. *For any function ensemble $\mathcal{H}$, the signature scheme $SIG^1_{\mathcal{G},\mathcal{H}}$ is insecure.*

Proof. We construct a forger that, on input any message M and any verification-key $VK = (PK', h)$ (where $PK' = (PK, f)$ and $h \in \mathcal{H}_n$) , generates a signature of M with respect to VK, as follows.

1. Let C be a circuit computing the hash function h. Let C_M be a circuit such that for every x, $C_M(x) =$ n most-significant-bits of $C(x, M)$.
2. Compute $w = auth_f(\hat{C}_M)$.
3. Compute the tree-commitment $a = TC_f(\hat{C}_M)$.
4. Compute $(b, q) = C(a, M)$.
5. Emulate the interaction $(P^{\mathcal{G}}(w), V^{\mathcal{G}}|_q)(f, a, b)$, to produce a transcript $(q, ans) \leftarrow (P^{\mathcal{G}}(w), V^{\mathcal{G}}|_q)(f, a, b)$.[12]
6. Output $(a, (b, q), ans)$.

It is trivial to verify that all forger steps are polynomial-time computable, and by completeness of $(P^{\mathcal{G}}, V^{\mathcal{G}})$, the forger will always be successful. $\square$

6.1.2 On the Security of ID^1

To establish $\neg(FS)$ it remains to show that there exists a function ensemble $\mathcal{G}$, such that $ID^1_{\mathcal{G}}$ is secure. Notice that it is easy to prove the security of $ID^1_{\mathcal{G}}$ under the CSP hypothesis.

Lemma 6.2. *Under the CSP hypothesis, there exists a function ensemble $\mathcal{G}$ such that $ID^1_{\mathcal{G}}$ is secure.*

Proof. The CSP hypothesis implies that there exists a function ensemble $\mathcal{G}$ for which $(P^{\mathcal{G}}, V^{\mathcal{G}})$ satisfies both the completeness property and the CS-proof-of-knowledge property (follows from Proposition 2). It is easy to verify that $ID^1_{\mathcal{G}}$ is secure, with respect to this function ensemble $\mathcal{G}$. $\square$

Thus, we proved $(CSP) \implies \neg(FS)$.

Unfortunately, we do not know how to prove (directly) $\neg(CSP) \implies \neg(FS)$. Instead we proceed as follows. Consider the following two cases.

[12] Note that $((f, a, b), w) \in \mathcal{R}_{\mathcal{F}}$.

- (Case 1): There exists a function ensemble $\mathcal{G}$ such that $ID^1_{\mathcal{G}}$, is secure.
- (Case 2): For every function ensemble $\mathcal{G}$, $ID^1_{\mathcal{G}}$ is not secure.

If we are in Case 1 we are done, since then there exists a function ensemble $\mathcal{G}$ such that $ID^1_{\mathcal{G}}$ is secure, whereas $FS_{\mathcal{H}}(ID^1_{\mathcal{G}})$ is insecure with respect to any function ensemble $\mathcal{H}$, and $\neg(FS)$ is established. Hence, we assume that we are in Case 2. Namely, we assume that for every function ensemble $\mathcal{G}$, there exists polynomial-size circuit family $F_1 = \{F_1^n\}$, a polynomial-size circuit family $\tilde{P}_1 = \{\tilde{P}_1^n\}$, and a polynomial $p(\cdot)$, such that for infinitely many n's,

$$Pr[(\tilde{P}_1^n, V^{\mathcal{G}})(f,a,b) = 1 \ : \ a = F_1^n(f)] \geq \frac{1}{p(n)}$$

(where the probability is over $f \in_R \mathcal{F}_n$, over $b \in_R \{0,1\}^n$ and over the random coin tosses of $V^{\mathcal{G}}$). We denote the set of all such n's by $S^1_{\mathcal{G}}$.

We refer to this case by ($\forall \mathcal{H}$ $\exists$ IMPERSONATOR)

It remains to prove the following lemma.

Lemma 6.3. *($\forall \mathcal{G}$ $\exists$ IMPERSONATOR) $\Rightarrow \neg(FS)$.*

To prove this Lemma we construct yet two more ID schemes ID^2 and ID^3, such that one of them demonstrates the failure of the FS paradigm.

6.2 Construction of ID^2

The assumption ($\forall \mathcal{G}$ $\exists$ IMPERSONATOR) implies in particular, that for every $n \in S^1_{\mathcal{G}}$, given a random $f \in \mathcal{F}_n$, it is easy to find a and $b_1 \neq b_2$, and to convince both $V^{\mathcal{G}}(f,a,b_1)$ and $V^{\mathcal{G}}(f,a,b_2)$, with non-negligible probability.

In contrast, it is hard to convince both $V^0(f,a,b_1)$ and $V^0(f,a,b_2)$, since (P^0,V^0) is a proof of knowledge, and anyone who knows a witness to both (f,a,b_1) and (f,a,b_2) can be used to find collisions to f.

This contrast between V^0 and $V^{\mathcal{G}}$ suggests constructing a new ID scheme, ID^2, whose security will follow from the proof-of-knowledge property of (P^0,V^0) on one hand, and on the other hand the insecurity of the corresponding digital signature scheme (obtained from the Fiat-Shamir paradigm) will follow from the assumption ($\forall \mathcal{G}$ $\exists$ IMPERSONATOR).

Let $\mathcal{F}$ be a collision resistant hash-function ensemble, and let $ID = (G,S,R)$ be any secure canonical ID scheme. We extend ID to obtain a new ID scheme $ID^2 = (G^2,S^2,R^2)$, by extending the public key and the verdict function, as follows.

- G^2: On input 1^n,

  1. Run $G(1^n)$, to obtain a pair $(SK, PK) \leftarrow G(1^n)$.
  2. Choose uniformly
     - $f, f_1^{UA}, f_2^{UA} \in \mathcal{F}_n$
     - $k \in KEY_n$ (a seed for $COMM$)
     - $r \in \{0,1\}^n$ (randomness for $COMM$)
     - γ_1' (randomness for V_{PCP}).

  Output SK as the secret-key and $PK' = (PK, f, (f_1^{UA}, f_2^{UA}), (k, r), \gamma_1')$ as the public-key.

- R^2: On input a public-key $PK' = (PK, f, (f_1^{UA}, f_2^{UA}), (k, r), \gamma_1')$, R^2 will accept either views that $R(PK)$ will accept or views of the form

$$\begin{array}{ccc} S^2 & & R^2 \\ & \xrightarrow[\hat{\beta}_2]{\hspace{6em}} & \\ & \xleftarrow[\gamma_1'', \gamma_2]{\hspace{7em}} & \\ & \xrightarrow[a, b_1, b_2, \beta_1, \beta_2, \delta_1, \delta_2]{\hspace{7em}} & \end{array}$$

  where

  - $(f_1^{UA}; \beta_1; \gamma_1' \oplus \gamma_1''; \delta_1) \in VIEW(V^0(f, a, b_1))$.
  - $(f_2^{UA}; \beta_2; \gamma_2; \delta_2) \in VIEW(V^0(f, a, b_2))$.
  - $\hat{\beta}_2$ commits to $a, b_1, b_2, \beta_1, \beta_2$, as follows

$$\hat{\beta}_2 = comm_k(\beta_2; comm_k(a, b_1, b_2, \beta_1; r)).$$

Intuitively, the above view can be thought of as an interleaved execution of the following two views:

$$\begin{array}{ccc}
P^0 & (f,a,b_1) & V^0 \\
 & \xleftarrow[f_1^{UA}]{\qquad\qquad} & \\
 & \xrightarrow[\beta_1]{\qquad\qquad} & \\
 & \xleftarrow[\gamma_1'' \oplus \gamma_1']{\qquad\qquad} & \\
 & \xrightarrow[\delta_1]{\qquad\qquad} &
\end{array}
\qquad
\begin{array}{ccc}
P^0 & (f,a,b_2) & V^0 \\
 & \xleftarrow[f_2^{UA}]{\qquad\qquad} & \\
 & \xrightarrow[\beta_2]{\qquad\qquad} & \\
 & \xleftarrow[\gamma_2]{\qquad\qquad} & \\
 & \xrightarrow[\delta_2]{\qquad\qquad} &
\end{array}$$

Remark: It is necessary to append γ_1' to the public-key in order to later establish the insecurity of the corresponding signature scheme. More specifically, when ID^2 will be converted into a signature scheme (by applying the Fiat-Shamir paradigm), the verifier will be replaced with a hash-function, and thus γ_1'' will no longer necessarily be chosen at random. Yet, we only know how to establish the insecurity of the signature scheme assuming that γ_1'' is chosen at random. We get around this problem by XORing γ_1'' with a uniformly distributed string γ_1', from the public-key.

6.2.1 The Security of ID^2

Lemma 6.4. *Assuming $\mathcal{F}$ is collision resistant, ID^2 is secure.*

Proof Idea: Assume for contradiction that ID^2 is not secure. That is, assume that there exists a cheating sender $\tilde{S} = \{\tilde{S}_n\}$ and a polynomial $p(\cdot)$ such that for infinitely many n's, $Pr[(\tilde{S}_n, R^2)(PK') = 1] \geq \frac{1}{p(n)}$ (where the probability is over $PK' \leftarrow G^2(1^n)$ and over the random coin tosses of R^2).

We will prove that the existence of $\tilde{S}$ implies the existence of a circuit that finds collisions in $\mathcal{F}$. This will be done in two parts, as follows.

- **(Part 1):** We will first show that there exist non-uniform probabilistic-polynomial-time Turing machines $F = \{F_n\}$ and $\tilde{P} = \{\tilde{P}_n\}$, such for infinitely many n's the following holds.

  For $(a, b_1, b_2, aux_1, aux_2) = F_n(f, f_1^{UA}, f_2^{UA})$,

  $$Pr\left[(\tilde{P}_n(aux_1), V^0|_{f_1^{UA}})(f,a,b_1) = 1 \;\wedge\; (\tilde{P}_n(aux_2), V^0|_{f_2^{UA}})(f,a,b_2) = 1\right] \geq 1/p(n)^3$$

(where the probability is over a uniformly chosen $f, f_1^{UA}, f_2^{UA} \in \mathcal{F}_n$, and over the random coin tosses of F_n, $\tilde{P}_n$, $V^0|_{f_1^{UA}}$ and $V^0|_{f_2^{UA}}$).[13]

The proof-of-knowledge property of (P^0, V^0) will imply that there exists a probabilistic-polynomial-time oracle machine E and a polynomial $p'(\cdot)$ such that for any $(a, b_1, b_2, aux_1, aux_2)$ which satisfy the above inequality,

$$Pr\left[\begin{array}{c} \forall i\ E^{\tilde{P}_n(aux_1)}((f,a,b_1),i) = w_i^1 \text{ s.t. } ((f,a,b_1),w^1) \in \mathcal{R}_{\mathcal{F}} \\ and \\ \forall i\ E^{\tilde{P}_n(aux_2)}((f,a,b_2),i) = w_i^2 \text{ s.t. } ((f,a,b_2),w^2) \in \mathcal{R}_{\mathcal{F}} \end{array}\right] \geq \frac{1}{p'(n)}$$

(where the probability is over the random coin tosses of $E^{\tilde{P}_n(aux_1)}$ and $E^{\tilde{P}_n(aux_2)}$).

- **(Part 2):** We will then show that there exists a probabilistic-polynomial-time oracle machine, with oracle access to E, F_n and $\tilde{P}_n$, such that, on input a uniformly chosen $f \in_R \mathcal{F}_n$, outputs a collision in f, with non-negligible probability.

Note that since non-uniform probabilistic-polynomial-time Turing machines can be modelled as polynomial-size circuits, Part 1 together with Part 2 imply the existence of a polynomial-size circuit such that, on input a uniformly chosen $f \in_R \mathcal{F}_n$, outputs a collision in f, with non-negligible probability. This will contradict the assumption that $\mathcal{F}$ is collision resistant.

The formal proof is quite tedious and is deferred to Appendix B.

We next consider the insecurity of the corresponding signature scheme. For every $\mathcal{H}$, we denote $FS_{\mathcal{H}}(ID^2)$ by $SIG^2_{\mathcal{H}} = (GEN^2_{\mathcal{H}}, SIGN^2_{\mathcal{H}}, VERIFY^2_{\mathcal{H}})$.

6.2.2 On the Insecurity of $SIG^2_{\mathcal{H}}$

Proving the insecurity of $SIG^2_{\mathcal{H}} = FS_{\mathcal{H}}(ID^2)$ is tricky. Intuitively, we would like to use the assumption ($\forall\mathcal{H}\exists$ IMPERSOPNATOR) to forge signatures, as follows. Fix $h \in \mathcal{H}$. Given a random verification key $VK = (PK, f, (f_1^{UA}, f_2^{UA}), (k, r), \gamma', h)$, use the IMPERSONATOR to find an a such that for random $b_1 \neq b_2$, the IMPERSONATOR can fool both $V^{\mathcal{H}}(f, a, b_1)$ and $V^{\mathcal{H}}(f, a, b_2)$ to accept. However, in this approach the IMPERSONATOR finds $\beta_1, \beta_2, \gamma_1, \gamma_2$ such that γ_1 depends only on β_1 and γ_2 depends only on β_2, whereas in valid signatures γ_1 and γ_2 are functions of both β_1 and β_2. Thus, to obtain a valid signature, we cannot simply run $\tilde{P}_1^n$ twice independently, since the value of β_2 affects the value of γ_1 and vise versa.

To get around this problem we distinguish between the following two cases:

[13] recall that $V^0|_{f^{UA}}$ is V^0, restricted to sending f^{UA} as the first message.

- *Case 2a:* ($\forall \mathcal{G}\ \exists$ *strong*-IMPERSONATOR)
- *Case 2b:* $\neg$($\forall \mathcal{G}\ \exists$ *strong*-IMPERSONATOR)

Where ($\forall \mathcal{G}\ \exists$ *strong*-IMPERSONATOR) refers to the case that for every function ensemble $\mathcal{G}$ there exists a "*strong*"-impersonator, that for infinitely many n's, on input a random $f \in \mathcal{F}_n$, finds a and b_1 such that he can convince $V^0(f, a, b_1)$ to accept and convince $V^{\mathcal{G}}(f, a, b_2)$ to accept for a random b_2. We denote the set of all such n's by $S^2_{\mathcal{G}}$. Formally speaking, ($\forall \mathcal{G}\ \exists$ *strong*-IMPERSONATOR) refers to the case that for every function ensemble $\mathcal{G}$ there exists a polynomial-size circuit family $F_2 = \{F_2^n\}$, a polynomial-size circuit family $\tilde{P}_2 = \{\tilde{P}_2^n\}$ and a polynomial $p(\cdot)$ such that for every $n \in S^2_{\mathcal{G}}$,

$$Pr[(\tilde{P}_2^n, V^0)(f, a, b_1) = 1 \quad \wedge \quad (\tilde{P}_2^n, V^{\mathcal{G}})(f, a, b_2) = 1 : \\ (a, b_1) = F_2^n(f)] \geq \frac{1}{p(n)}$$

(where the probability is over $f \in_R \mathcal{F}_n$, over $b_2 \in_R \{0,1\}^n$ and over the random coin tosses of $V^{\mathcal{G}}$ and V^0).

We proceed by proving the failure of the FS paradigm is case 2a and in case 2b.

The Failure of the FS Paradigm in Case 2a:

In this case, we proceed with ID^2 and show that $SIG^2_{\mathcal{H}}$ is insecure for every $\mathcal{H}$, and for every $n \in S^2_{\mathcal{G}}$.

Lemma 6.5. *Assuming ($\forall \mathcal{H}\ \exists$ strong-IMPERSONATOR), for any function ensemble $\mathcal{H}$ the signature scheme $SIG^2_{\mathcal{H}}$ is insecure.*

Proof. Fix a function ensemble $\mathcal{H}$. We show that for every message M there exists a forger $FORG^M$ which, on input a random verification-key VK, outputs a signature of M, with non-negligible probability. Fix any message M. For any $n \in \mathbb{N}$ and for any $h \in \mathcal{H}$, define $h^M(x) =$ n least-significant-bits of $h(x, M)$, and let $\mathcal{H}^M = \{h^M\}_{h \in \mathcal{H}}$. From our assumption there exist two polynomial-size circuit families $F_2 = \{F_2^n\}_{n \in \mathbb{N}}$ and $\tilde{P}_2 = \{\tilde{P}_2^n\}$ such that for every $n \in S^2_{\mathcal{H}^M}$ and for every $(a, b_1) = F_2^n(f)$,

$$Pr[(\tilde{P}_2^n, V^0)(f, a, b_1) = 1 \ \wedge \ (\tilde{P}_2^n, V^{\mathcal{H}^M})(f, a, b_2) = 1] \geq \frac{1}{poly}$$

(where the probability is over $f \in_R \mathcal{F}_n$, $b_2 \in_R \{0,1\}^n$ and the random coin tosses of V^0 and $V^{\mathcal{H}^M}$).

On input a random verification-key $VK = (PK', h)$, where $h \in \mathcal{H}_n$ and $PK' = (PK, f, (f_1^{UA}, f_2^{UA}), (k, r), \gamma_1')$, the forger $FORG^M$ generates a signature of M as follows.

1. Compute $(a, b_1) = F_2^n(f)$.
2. Emulate the interaction of $(\tilde{P}_2^n, V^0|_{f_1^{UA}})(f, a, b_1)$, to obtain a transcript
$$(f_1^{UA}; \beta_1; *; *) \leftarrow (\tilde{P}_2^n, V^0|_{f_1^{UA}})(f, a, b_1).$$
3. Choose randomly $b_2 \in \{0,1\}^n$, and let $r' = comm_k(a, b_1, b_2, \beta_1; r)$.
4. Choose randomly $h_2 \ldots, h_n \in \mathcal{H}_n$, and let
$$q_M = (f_2^{UA}, (k, r'), (h^M, h_2^M \ldots, h_n^M)).$$
5. Emulate the interaction of $(\tilde{P}_2^n, V^{\mathcal{H}^M}|_{q_M})(f, a, b_2)$, to obtain a transcript
$$(q_M; ans) \leftarrow (\tilde{P}_2^n, V^{\mathcal{H}^M}|_{q_M})(f, a, b_2).$$
Denote $ans = \{\beta_2^i, \hat{\beta}_2^i, \gamma_2^i, \delta_2^i\}_{i=1}^n$.
6. Compute $(\gamma_1'', *) = h(\hat{\beta}_2^1, M)$.
7. Emulate the interaction $(\tilde{P}_2^n, V^0|_{f_1^{UA}, \gamma_1' \oplus \gamma_1''})(f, a, b_1)$ to obtain a transcript
$$(f_1^{UA}; \beta_1; \gamma_1' \oplus \gamma_1''; \delta_1) \leftarrow (\tilde{P}_2^n, V^0|_{f_1^{UA}, \gamma_1' \oplus \gamma_1''})(f, a, b_1).$$
8. Output $(\hat{\beta}_2; (\gamma_1'', \gamma_2); (a, b_1, b_2, \beta_1, \beta_2, \delta_1, \delta_2))$ as a signature of M.

We claim that the forger will be successful with non-negligible probability.

Claim 6.5.1. *$Pr[VERIFY_{\mathcal{H}}^2(VK, M, FORG^M(VK)) = 1] = non\text{-}negl(n)$ (where the probability is over VK and over the random coin tosses of $FORG^M$).*

Since the proof is quite technical it is deferred to Appendix C.

□

It remains to prove the failure of the FS paradigm in case $2b$. We construct yet another and final ID scheme ID^3, which will demonstrate the failure of the FS paradigm in this case.

6.3 Construction of ID^3

Throughout this subsection we assume

$$(\forall \mathcal{G}\ \exists\ \text{IMPERSONATOR}) \ \wedge\ \neg(\forall \mathcal{G}\ \exists \textit{strong}\text{-IMPERSONATOR}) \ \Rightarrow\ \neg(\text{FS})$$

We establish $\neg(FS)$ by extending any secure ID scheme into a new ID scheme $ID^3 = (G^3, S^3, R^3)$. The security of ID^3 will follow from the assumption $\neg(\forall \mathcal{G}\ \exists\ \textit{strong}\text{-IMPERSONATOR})$, and the insecurity of the corresponding signature scheme $SIG^3_{\mathcal{H}} = FS_{\mathcal{H}}(ID^3)$ (for $n \in S^1_{\mathcal{H}}$) will follow from the assumption $(\forall \mathcal{G}\ \exists\ \text{IMPERSONATOR})$.

Recall that, roughly speaking, in ID^1 there was one execution of $(P^{\mathcal{G}}, V^{\mathcal{G}})$. In ID^2 there were two parallel executions of (P^0, V^0). ID^3 will be in some sense a hybrid of ID^1 and ID^2. It will once execute $(P^{\mathcal{G}}, V^{\mathcal{G}})$ and once execute (P^0, V^0).

Fix a hash-function ensemble $\mathcal{G}$ that does not have a *strong*-IMPERSONATOR (one exists by assumption). Take any secure canonical ID scheme $ID = (G, S, R)$ and define ID^3 as follows.

- G^3: On input 1^n,
  1. Run $G(1^n)$, to obtain a pair $(SK, PK) \leftarrow G(1^n)$.
  2. Choose uniformly
     - $f, f^{UA} \in \mathcal{F}_n$
     - $k \in KEY_n$ (a key for $COMM$)
     - $r \in \{0,1\}^n$ (randomness for $COMM$)
     - $b'_2 \in \{0,1\}^n$
     - q' (a first message sent by $V^{\mathcal{H}^1}$).

  Output SK as the secret-key and $PK' = (PK, f, f^{UA}, (k, r), (b'_2, q'))$ as the public-key.

- R^3: On input a public-key $PK' = (PK, f, f^{UA}, (k, r), (b'_2, q'))$, R^3 accepts either views that $R(PK)$ accepts or views of the form

$$\begin{array}{ccc} S^3 & & R^3 \\ & \xrightarrow[\hat{\beta}_1]{\qquad\qquad} & \\ & \xleftarrow[\gamma_1, (b''_2, q'')]{} & \\ & \xrightarrow[a, b_1, \beta_1, \delta_1, ans]{} & \end{array}$$

  where

- $(f^{UA};\beta_1;\gamma_1;\delta_1) \in VIEW(V^0(f,a,b_1))$
- $(q' \oplus q''; ans) \in VIEW(V^{\mathcal{G}}(f,a,b_2' \oplus b_2''))$
- $\hat{\beta_1} = comm_k(\beta_1; comm_k(a,b_1;r))$.

Intuitively, the above view can be thought of as an interleaved execution of the following two views:

$$\begin{array}{ccc} P^0 & (f,a,b_1) & V^0 \\ & \xleftarrow[f^{UA}]{\qquad} & \\ & \xrightarrow[\beta_1]{\qquad} & \\ & \xleftarrow[\gamma_1]{\qquad} & \\ & \xrightarrow[\delta_1]{\qquad} & \end{array} \qquad \begin{array}{ccc} P^{\mathcal{G}} & (f,a,b_2' \oplus b_2'') & V^{\mathcal{G}} \\ & \xleftarrow[q' \oplus q'']{\qquad} & \\ & \xrightarrow[ans]{\qquad} & \end{array}$$

Remark: It is necessary to append b_2', q' to the public-key in order to later establish the insecurity of $FS_{\mathcal{H}}(ID^3)$. More specifically, when ID^3 will be converted into a signature scheme (by applying the Fiat-Shamir paradigm), the verifier will be replaced with a hash function, and thus b_2'' and q'' will no longer necessarily be chosen at random. Yet, we only know how to establish the insecurity of the signature scheme assuming that b_2'' and q'' are chosen at random. We get around this problem by XORing b_2'' with a uniformly distributed string b_1' and XORing q'' with a uniformly distributed string q'.

Lemma 6.6. *Assuming $\mathcal{G}$ does not have a strong-IMPERSONATOR, ID^3 is secure.*

Proof. Follows easily from the definition of a *strong*-IMPERSONATOR. □

We denote $FS_{\mathcal{H}}(ID^3)$ by $SIG^3_{\mathcal{H}} = (GEN^3_{\mathcal{H}}, SIGN^3_{\mathcal{H}}, VERIFY^3_{\mathcal{H}})$.

Lemma 6.7. *Assuming ($\forall \mathcal{H}\ \exists IMPERSONATOR$), for any function ensemble $\mathcal{H}$ the signature scheme $SIG^3_{\mathcal{H}}$ is insecure.*

To prove the insecurity of $SIG^3_{\mathcal{H}}$, fix any message M. We want to exhibit a forgery of M. The crux of the idea is that in order to produce a valid signature for M it suffices to find a, b_1, and to carry out two reduced-interaction universal arguments, one for (f,a,b_1) and one for (f,a,b_2). It seems like this could be done using our friend IMPERSONATOR. However, there is subtle point here. The reduced-interaction universal argument for (f,a,b_2) is carried out with $V^{\mathcal{G}}(f,a,b_2)$, and so we would like to use an IMPERSONATOR for $V^{\mathcal{G}}$, whereas the reduced-interaction universal argument for (f,a,b_1) is carried out with $V^{\mathcal{H}^M}(f,a,b_1)$, where $\mathcal{H}^M$ a function ensemble which is defined as follows: For any $n \in \mathbb{N}$ and for any $h \in \mathcal{H}_n$, define $h^M(x) =$ n most-significant-bits of $h(x,M)$, and let $\mathcal{H}^M = \{h^M\}_{h\in\mathcal{H}}$.

Thus, it seems like we need to use two different IMPERSONATORS, one for $\mathcal{G}$ and one for $\mathcal{H}^M$. However, the problem is that the IMPERSONATOR for $\mathcal{H}^M$ and the IMPERSONATOR for $\mathcal{G}$ may impersonate with respect to different a's. We get around this problem by using a single IMPERSONATOR for $\mathcal{H}' = \mathcal{H}^M \cup \mathcal{G}$. Details follow.

Proof. Fix a function ensemble $\mathcal{H}$. We exhibit a forger for $SIG^3_{\mathcal{H}}$. More specifically, we show that for every message M there exists a forger $FORG^M$ which, on input a random verification key VK, outputs a signature of M, with non-negligible probability.

Fix any message M, and define $\mathcal{H}^M$ and $\mathcal{H}'$ as above. By our assumption ($\forall\mathcal{H}$ $\exists$IMPERSONATOR), there exist $F_1 = \{F_1^n\}_{n\in\mathbb{N}}$, $\tilde{P}_1 = \{\tilde{P}_1^n\}$, and a polynomial $p(\cdot)$ such that for every $n \in S^1_{\mathcal{H}'}$ and for $a = F_1^n(f)$,

$$Pr[(\tilde{P}_1^n, V^{\mathcal{H}'})(f,a,b) = 1] \geq \frac{1}{p(n)}$$

(where the probability is over $f \in_R \mathcal{F}_n$, $b \in_R \{0,1\}^n$ and the random coin tosses of $V^{\mathcal{H}'}$).

Claim 6.7.1. *There exists a polynomial-size circuit $\tilde{\tilde{P}}_1^n$ and a polynomial $p'(\cdot)$ such that for every $n \in S^1_{\mathcal{H}'}$ and for $a = F_1^n(f)$,*

$$Pr[(\tilde{\tilde{P}}_1^n, V^{\mathcal{G}})(f,a,b_1) = 1 \ \wedge \ (\tilde{\tilde{P}}_1^n, V^{\mathcal{H}^M})(f,a,b_2) = 1] \geq \frac{1}{p'(n)}$$

(where the probability is over $f \in_R \mathcal{F}_n$, $b_1, b_2 \in_R \{0,1\}^n$ and the random coin tosses of $V^{\mathcal{G}}$ and $V^{\mathcal{H}^M}$).

Again, due to the technical nature of the proof it is deferred to Appendix D.

We are now ready to exhibit the forger $FORG^M$. To simplify notations, from now on we denote $\tilde{\tilde{P}}$ by $\tilde{P}$.
On input a verification-key $VK = (PK', h)$, where $h \in \mathcal{H}_n$ and $PK' = (PK, f, f^{UA}, (k,r), (b_2', q'))$, The forger $FORG^M$ generates a signature of M, with respect to VK, as follows.

1. Compute $a = F_1^n(f)$.

2. (a) Choose $b_1 \in_R \{0,1\}^n$, and compute $r' = comm_k(a, b_1; r)$.

   (b) Choose $h_2, \ldots, h_n \in_R \mathcal{H}_n^M$, and set

$$q^M = (f^{UA}, (k, r'), (h^M, h_2^M, \ldots, h_n^M)).$$

(c) Emulate the interaction of $(\tilde{P}_1^n, V^{\mathcal{H}^M}|_{q^M})(f, a, b_1)$ to obtain a transcript

$$(q^M; ans^M) \leftarrow (\tilde{P}_1^n, V^{\mathcal{H}^M}|_{q^M})(f, a, b_1).$$

Denote $ans^M = \{\beta_i, \hat{\beta}_i, \gamma_i, \delta_i\}_{i=1}^n$.

3. Compute $(*, (b_2'', q'')) = h(\hat{\beta}_1, M)$.

4. Emulate the interaction of $(\tilde{\tilde{P}}_n^1, V^{\mathcal{G}}|_{q' \oplus q''})(f, a, b_2' \oplus b_2'')$, to obtain a transcript

$$(q' \oplus q''; ans) \leftarrow (\tilde{\tilde{P}}_n^1, V^{\mathcal{G}}|_{q' \oplus q''})(f, a, b_2' \oplus b_2'').$$

5. Output $(\hat{\beta}_1, (\gamma_1, (b_2'', q'')), (a, b_1, \beta_1, \delta_1, ans))$ as a signature of M.

We claim that the forger will be successful with non-negligible probability.

Claim 6.7.2. *There exists a polynomial $p(\cdot)$ such that for every $n \in S_{\mathcal{H}'}^1$*

$$Pr[VERIFY_{\mathcal{H}}^3(VK, M, FORG^M(VK)) = 1] \geq \frac{1}{p(n)}$$

(where the probability is over VK and over the random coin tosses of $FORG^M$).

Again, due to the technical flavor of the proof of the above claim, we defer it to Appendix E.

Thus, we have established the insecurity of $SIG_{\mathcal{H}}^3$. □

Figure 1 summarizes the outline of the proof of Theorem 2.

7 On the Failure of FS Modifications

The FS paradigm was designed for constructing signature schemes by eliminating interaction from canonical ID schemes. We proved that this paradigm fails in the sense that there exist secure canonical ID schemes for which the corresponding signature scheme (obtained by the FS method) is insecure with respect to any function ensemble. A question that remains is: Do there exist other *secure* methods for eliminating interaction?

Two modifications of the FS paradigm were considered in the literature: One due to Micali and Reyzin [MR02] and the other due to Abdalla, An, Bellare and Nampremre [AABN02]. Using similar ideas to the ones presented in this paper, one can prove the failure of these FS modifications as well.

7.1 First Modification

Micali and Reyzin [MR02] presented a method for constructing FS-like signature schemes that yield better "exact security" than the original FS method. In their method, they convert any ID scheme $(\alpha; \beta; \gamma)$ into a signature scheme, in which the signer first chooses β and only then produces α by computing $\alpha = h(\beta, M)$, where M is the message to be signed and h is the function used to reduce interaction.[14]

We argue that this FS-like method proposed in [MR02] is insecure, as follows. Take any secure ID scheme and modify it by appending $f \in_R \mathcal{F}$ to the public key, and extending its verdict function so as to also accept views of the following form

$$\begin{array}{ccc} S & (PK, f) & R \\ & \xrightarrow[a]{\hspace{2cm}} & \\ & \xleftarrow[b, q]{\hspace{2cm}} & \\ & \xrightarrow[ans]{\hspace{2cm}} & \end{array}$$

where

$$(q; ans) \in VIEW(V^{\mathcal{H}}(f, (b, q), a)).$$

We denote this extended ID scheme by $ID_{\mathcal{H}}$. It is relatively easy to show that the signature scheme, obtained by applying the above FS-like method to $ID_{\mathcal{H}}$, is insecure with respect to any function ensemble. Thus, if there exists a function ensemble $\mathcal{H}$ such that $ID_{\mathcal{H}}$ is secure, then the aove FS-like method is insecure. Namely, under the CSP hypothesis, the above FS-like method is insecure. To complete the proof one needs to assume that for every function ensemble $\mathcal{H}$, $ID_{\mathcal{H}}$ is insecure. The rest of the proof is quite technical and follows the lines of Sections 6.2 and 6.3.

7.2 Second Modification

Abdalla et. al. defined a randomized generalization of the FS paradigm, and showed that signature schemes, obtained from the generalized FS paradigm, are secure (resp. forward secure) in the Random Oracle Model if and only if the underlying ID scheme is secure (resp. forward secure) against impersonation under passive attacks. Their randomized method transforms

[14] Note that this method can be applied only to ID schemes in which the sender can compute γ only given (SK, PK, α, β), and does not need any additional information on α.

any canonical ID scheme $(\alpha;\beta;\gamma)$ into a signature scheme by replacing the random β with $h(\alpha, M, R)$, where M is the message to be signed, h is the function used to reduce interaction, and R is randomness chosen by the signer.

The failure of this generalized FS paradigm follows trivially from the fact that it is a generalization of the original FS paradigm with $R = \emptyset$ and from the fact that the original FS paradigm fails.

8 Future Directions

We have shown examples of digital signature schemes, that are obtained from secure identification schemes by applying the Fiat-Shamir Paradigm, and are insecure regardless of which "hash" function is used. Several related questions arise.

1. Our proof does not imply that the ID schemes used in practice such as [FFS88] or [Sch91] combined with some particular hash function ensemble H necessarily yield insecure digital signature schemes. It does imply that a proof of security would have to involve the particulars of the ID scheme and the H in question. Can one exhibit a proof of security (based on standard intractability assumptions) of $FS_{\mathcal{H}}(ID)$ for *any* practiced ID scheme and *any* H.

2. We showed that the FS paradigm and its known modifications [MR02, AABN02] fail. But, perhaps there exists another general efficient transformation from secure interactive ID schemes to digital signature schemes which can be proven secure?

3. Do there exist other "natural" cryptographic practices which are secure in the Random Oracle Model, and become insecure when the random oracle is replaced with any public function (chosen at random from some function ensemble)? Many examples of such "natural" practices exist for which no evidence of security exists outside the Random Oracle Model.

   In particular, an example that we are interested in is the non-interactive CS-proofs, constructed by Micali [Mi94] in the Random Oracle Model. Does there exist a language L for which there is no function ensemble $\mathcal{H}$ (replacing the Random Oracle), for which CS-proofs for L remain sound (or remain a proof-of-knowledge).

4. In the ID schemes which we constructed to demonstrate the failure of the FS paradigm, soundness is based on the prover being computationally bounded (i.e it is an argument rather than an interactive proof). Can one show that the Fiat-Shamir paradigm fails for an ID scheme for which soundness holds unconditionally? Note that [FS86] is of this latter type, whereas [Sch91] is an argument.

5. Our proof technique can be viewed as a way to reduce interaction in argument systems while preserving some security properties. Can this be extended to show that there exist 3 round zero knowledge arguments? Currently it is known that using the black box zero-knowledge definition they do not exist.

9 Acknowledgements

We are grateful to Chun-Yun Hsiao, Oded Goldreich, and Ran Canetti for useful comments on this work.

References

[AABN02] M. Abdalla, J. An, M. Bellare and C. Namprempre. From identification to signatures via the Fiat-Shamir transform: minimizing assumptions for security and forward-security. *Advances in Cryptography-EUROCRYPT 02, Lecture Notes in Computer Science, Springer-Verlag*, 2002.

[Bar01] B. Barak. How to go beyond the black-box simuation barrier. In *Proc. of the 42nd FOCS*, 2001.

[BF01] D. Boneh and M. Franklin. Identity-based encryption from Weil pairing. Preliminary version in *Crypto*, 2001.

[BG01] B. Barak and O. Goldreich. Universal arguments and their applications. *Proceedings of the 17th IEEE Annual Conference on Computational Complexity*, 2002.

[BGI$^+$01] B. Barak, O. Goldreich, R. Rudich, A. Sahai, S. Vadhan, and K.Yang. On the (im)possibility of obfuscating programs. In *Crypto* 2001.

[BM84] M. Blum, S. Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *SIAM J. Comput. 13(4): 850-864* 1984.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *Proceedings of the First Annual Conference on Computer and Communications Security.* ACM, November 1993.

[CGH98] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 209-218, Dalllas, 23-26 May, 1998.

[CS99] R. Cramer and V. Shoup. Signature schemes based on the strong RSA assumption. In *5th ACM Conference on Computer and Communications Security*, pages 46-51. Singapore, Nov. 1999. ACM Press.

[DH76] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22 (Nov.), pages 644-654, 1976.

[DNRS99] C. Dwork, M. Naor, O. Reingold and L.Stockmeyer. Magic funcitons. In *IEEE, editor, 40th Annual Symposium of Foundations of Computer Science*: October 17-19, 1999, New York City, New York, pages 523-534. *IEEE Computer Society Press*, 1999.

[FFS88] U. Feige, A. Fiat, and A. Shamir. Zero knowledge proofs of identity. *Journal of Cryptology*, 1(2), pp. 77-94, 1988.

[FS86] Amos Fiat and Adi Shamir. How to prove to yourself: practical solutions to identification and signature problems. In *Advances in Cryptology–Crypto 86,* pages 186-194, Springer, Berlin, 1987.

[Gol01] Oded Goldreich. Foundations of Cryptography, volume 1 – Basic Tools. *Cambridge University Press*, 2001.

[GHR99] R. Gennaro, S. Halevi and T. Rabin. Secure hash-and-sign signatures without the random oracle. *Advances in Cryptology - EUROCRYPT 99*, Lecture Notes in Computer Science Vol. 1592, J. Stern ed., Springer-Verlag, 1999.

[GGM86] Oded Goldreich, Shafi Goldwasser and Silvio Micali. How to construct random functions. *Journal of the Association of Computing Machinery*, 33(4): 792-807, 1986.

[GMR88] S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal of Computing*, 17(2):281-308, April 1988.

[GQ88] L. Guillou and J. J. Quisquater. A "paradoxical" identity-based signature scheme resulting from zero-knowledge. *Advances in Cryptology-CRYPTO 88*, Lecture Notes in Computer Science Vol. 403, S. Goldwasser ed., Springer- Verlag, 1988.

[Ki92] J. Kilian. A note on efficient zero-knowledge proofs and arguments. In *24'th STOC*, pages 723-732, 1992.

[Mer90] R.C. Merkle. A certified digital signature. *Proceedings on Advances in Cryptology*, pages 218-238, July 1989, Santa- Barbara, California.

[Ok92] T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. *Advances in Cryptology - CRYPTO 92*, Lecture Notes in Computer Science Vol. 740, E. Brickell ed., Springer-Verlag, 1992.

[MD5] R. Rivest. The MD5 message-digest algorithm. *RFC* 1321, April 1992.

[Mi94] Silvio Micali. Computationally sound proofs. *SICOMP*, vol. 30(4), pages 1253-1298, 2000. Preliminary version in *35th FOCS*, 1994.

[MR02] S. Micali and L. Reyzin. Improving the exact security of digital signature schemes. *Jornal of Cryptology*, 15(1):1-18, 2002.

[Na91] M. Naor. Bit commitment using pseudorandom generators. *Journal of Cryptology*, Vol.4, pages 151-158, 1991.

[NY89] M. Naor and M. Yung. Universal one-way hash functions and their cryptogracphic applications. *STOC 89*.

[PS96] D. Pointcheval and J. Stern. Security proofs for signature schemes. In *Advances in Cryptology-EUROCRYPT 96*, vol.1070 of Lecutre Notes in Compute Science, pages 387-398. Springer-Verlag, 1996.

[Rom90] John Rompel: One-Way Functions are Necessary and Sufficient for Secure Signatures. STOC 1990: 387-394.

[Sch91] Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology* 4(3):161-174.

A Commitment Schemes

Naor [Na91] proved that commitment schemes exist assuming the existence of one-way functions. Namely, assuming the existence of one-way functions, there exists functions $l(n)$ and $t(n)$, which are polynomially related to n, and there exists a commitment scheme $COMMIT$ such that for every $n \in \mathbb{N}$ and for every $k \in KEY_n$, $commit_k : \{0,1\}^n \times \{0,1\}^{l(n)} \rightarrow \{0,1\}^{t(n)}$.

Proposition 3. *Assuming the existence of collision resistant hash-function ensembles, For any function $m(n)$, which is polynomially-related to n, there exists a commitment scheme $COMM$, with a corresponding set of keys KEY', such that for every $n \in \mathbb{N}$ and for every $k' \in KEY'_n$, $comm_{k'} : \{0,1\}^{m(n)} \times \{0,1\}^n \rightarrow \{0,1\}^n$.*

Proof. Let $\mathcal{F}^m$ be a collision resistant hash-function ensemble. Fix any function $m(n)$, which is polynomially-related to n. We assume without loss of generality that for every $n \in \mathbb{N}$ and for every $f_n \in \mathcal{F}_n$, $f_n : \{0,1\}^{m(n)} \rightarrow \{0,1\}^n$. Similarly, it is easy to verify that for any function $t(n)$, which is polynomially-related to n, there exists collision resistant hash-function ensemble $\mathcal{F}'$ such that

1. for every $n \in \mathbb{N}$ and for every $f'_n \in \mathcal{F}'_n$, $f'_n : \{0,1\}^{t(n)} \rightarrow \{0,1\}^n$.

2. for every $n \in \mathbb{N}$, $f^t_n(U_{t(n)}) \cong U_n$.

The set of keys for $COMM$ is defined as follows: For every $n \in \mathbb{N}$,

$$KEY'_n = \{(k, f_n, f'_n) : \; k \in KEY_n, \; f_n \in \mathcal{F}_n, \; , f'_n \in \mathcal{F}'_n\}.$$

For every $n \in \mathbb{N}$, every $(k, f_n, f'_n) \in KEY'_n$ and every $(x, r) \in \{0,1\}^{m(n)} \times \{0,1\}^n$, define

$$comm_{(k,f_n,f'_n)}(x; r) = f'_n(commit_k(f_n(x); g(r)),$$

where $g : \{0,1\}^n \rightarrow \{0,1\}^{l(n)}$ is a one-way pseudorandom generator.[15]

$COMM$ is computationally-hiding since

1. g is a pseudorandom generator

2. $COMMIT$ is computationally-hiding

3. $f'_n(U_{t(n)}) \cong U_n$.

$COMM$ is computationally-binding since

1. $\mathcal{F}'$ is a collision-resistance hash-function ensemble

2. $COMMIT$ is computationally-binding

3. $\mathcal{F}$ is a collision-resistance hash-function ensemble.

4. g is one-way.

□

[15] It was proven in [GGM86] that one-way pseudorandom generators exist assuming the existence of one-way functions.

B Proof of Lemma 6.4

Proof. Assume for contradiction that ID^2 is not secure. That is, assume that there exists a cheating sender $\tilde{S} = \{\tilde{S}_n\}$ and a polynomial $p(\cdot)$ such that for infinitely many n's, $Pr[(\tilde{S}_n, R^2)(PK') = 1] \geq \frac{1}{p(n)}$ (where the probability is over $PK' \leftarrow G^2(1^n)$ and over the random coin tosses of R^2).

Proof Plan: We will prove that the existence of $\tilde{S}$ implies the existence of a circuit that finds collisions in $\mathcal{F}$. This will be done in two parts, as follows.

- **(Part 1):** We will first show that there exist non-uniform probabilistic-polynomial-time Turing machines $F = \{F_n\}$ and $\tilde{P} = \{\tilde{P}_n\}$, such for infinitely many n's the following holds.

  For $(a, b_1, b_2, aux_1, aux_2) = F_n(f, f_1^{UA}, f_2^{UA})$,

  $$Pr\left[(\tilde{P}_n(aux_1), V^0|_{f_1^{UA}})(f, a, b_1) = 1 \ \wedge \ (\tilde{P}_n(aux_2), V^0|_{f_2^{UA}})(f, a, b_2) = 1\right] \geq 1/p(n)^3$$

  (where the probability is over a uniformly chosen $f, f_1^{UA}, f_2^{UA} \in \mathcal{F}_n$, and over the random coin tosses of F_n, $\tilde{P}_n$, $V^0|_{f_1^{UA}}$ and $V^0|_{f_2^{UA}}$).[16]
  The proof-of-knowledge property of (P^0, V^0) will imply that there exists a probabilistic-polynomial-time oracle machine E and a polynomial $p'(\cdot)$ such that for any $(a, b_1, b_2, aux_1, aux_2)$ which satisfy the above inequality,

  $$Pr\left[\begin{array}{c} \forall i \ E^{\tilde{P}_n(aux_1)}((f, a, b_1), i) = w_i^1 \text{ s.t. } ((f, a, b_1), w^1) \in \mathcal{R}_{\mathcal{F}} \\ \textit{and} \\ \forall i \ E^{\tilde{P}_n(aux_2)}((f, a, b_2), i) = w_i^2 \text{ s.t. } ((f, a, b_2), w^2) \in \mathcal{R}_{\mathcal{F}} \end{array}\right] \geq \frac{1}{p'(n)}$$

  (where the probability is over the random coin tosses of $E^{\tilde{P}_n(aux_1)}$ and $E^{\tilde{P}_n(aux_2)}$).

- **(Part 2):** We will then show that there exists a probabilistic-polynomial-time oracle machine, with oracle access to E, F_n and $\tilde{P}_n$, such that, on input a uniformly chosen $f \in_R \mathcal{F}_n$, outputs a collision in f, with non-negligible probability.

Note that since non-uniform probabilistic-polynomial-time Turing machines can be modelled as polynomial-size circuits, Part 1 together with Part 2 imply the existence of a polynomial-size circuit such that, on input a uniformly chosen $f \in_R \mathcal{F}_n$, outputs a collision in f, with

[16] recall that $V^0|_{f^{UA}}$ is V^0, restricted to sending f^{UA} as the first message.

non-negligible probability. This will contradict the assumption that $\mathcal{F}$ is collision resistant.

We proceed to carry out the proof plan.

Part 1:

- $F_n(f, f_1^{UA}, f_2^{UA})$ operates as follows.

  1. Choose uniformly
     - $PK \leftarrow G(1^n)$
     - $k \in KEY_n$ (a key for $COMM_n$)
     - $r \in \{0,1\}^n$ (randomness for $COMM_n$)
     - γ_1' (randomness for V_{PCP})

     and set $PK' = (PK, f, (f_1^{UA}, f_2^{UA}), (k, r), \gamma_1')$.
  2. Emulate an interaction of $(\tilde{S}_n, R^2)(PK')$ to obtain a transcript

     $$(\hat{\beta}_2; (\gamma_1'', \gamma_2); (a, b_1, b_2, \beta_1, \beta_2, \delta_1, \delta_2)) \leftarrow (\tilde{S}_n, R^2)(PK').$$
  3. Set $aux_1 = (\beta_1, PK')$ and $aux_2 = (\beta_2, PK')$.

  Output $(a, b_1, b_2, aux_1, aux_2)$.

- $\tilde{P}_n(aux_1)$, where $aux_1 = (\beta_1, PK')$, interacts with $V^0|_{f_1^{UA}}(f, a, b_1)$ as follows.

  - V^0 sends f_1^{UA} to $\tilde{P}_n$.
  - $\tilde{P}_n$ sends β_1 to V^0.
  - V^0 chooses γ_1^1 at random, and sends γ_1^1 to $\tilde{P}_n$.
  - $\tilde{P}_n$ chooses γ_2^1 at random and emulates the interaction of $(\tilde{S}_n, R^2|_{\gamma_1^1 \oplus \gamma_1', \gamma_2^1})(PK')$, to obtain a transcript

    $$(\hat{\beta}_2; (\gamma_1^1 \oplus \gamma_1', \gamma_2^1); (a', b_1', b_2', \beta_1', \beta_2', \delta_1', \delta_2')) \leftarrow (\tilde{S}_n, R^2|_{\gamma_1^1 \oplus \gamma_1', \gamma_2^1})(PK').$$

    $\tilde{P}_n$ sends δ_1' to V^0.

- $\tilde{P}_n(aux_2)$, where $aux_2 = (\beta_2, PK')$, interacts with $V^0|_{f_2^{UA}}(f, a, b_2)$ as follows.

  - V^0 sends f_2^{UA} to $\tilde{P}_n$.

- $\tilde{P}_n$ sends β_2 to V^0.
- V^0 chooses γ_2^2 at random and sends γ_2^2 to $\tilde{P}_n$.
- $\tilde{P}_n$ chooses γ_1^2 at random and emulates the interaction of $(\tilde{S}_n, R^2|_{\gamma_1^2,\gamma_2^2})(PK')$ to obtain a transcript

  $$(\hat{\beta}_2; (\gamma_1^2, \gamma_2^2); (a'', b_1'', b_2'', \beta_1'', \beta_2'', \delta_1'', \delta_2'')) \leftarrow (\tilde{S}_n, R^2|_{\gamma_1^2,\gamma_2^2})(PK').$$

  $\tilde{P}_n$ sends δ_2'' to V^0.

Claim B.0.3. *Let $F_n(f, f_1^{UA}, f_2^{UA}) = (a, b_1, b_2, aux_1, aux_2)$. Then, for infinitely many n's*

$$Pr\left[(\tilde{P}_n(aux_1), V^0|_{f_1^{UA}})(f, a, b_1) = 1 \ \wedge (\tilde{P}_n(aux_2), V^0|_{f_2^{UA}})(f, a, b_2) = 1\right] \geq 1/p(n)^3$$

(where the probability is over $f, f_1^{UA}, f_2^{UA} \in_R \mathcal{F}_n$, and over the random coin tosses of $V^0|_{f_1^{UA}}$ and $V^0|_{f_2^{UA}}$).

Proof. By the assumption made for contradiction, for infinitely many n's

$$Pr[(\tilde{S}_n, R^2)(PK') = 1] \geq 1/p(n)$$

(where the probability is over PK' and over the random coin tosses of R^2).
The fact that $\gamma_1'', \gamma_2, \gamma_1^1 \oplus \gamma_1', \gamma_2^1, \gamma_1^2, \gamma_2^2$ are all uniformly distributed and independent of PK', implies that for infinitely many n's, the following three conditions hold with probability at least $1/p(n)^3$.

- $(\tilde{S}_n, R^2|_{\gamma_1'',\gamma_2})(PK') = 1$
- $(\tilde{S}_n, R^2|_{\gamma_1^1\oplus\gamma_1',\gamma_2^1})(PK') = 1$
- $(\tilde{S}_n, R^2|_{\gamma_1^2,\gamma_2^2})(PK') = 1$

In other words,

- $(\hat{\beta}_2; (\gamma_1'', \gamma_2); (a, b_1, b_2, \beta_1, \beta_2, \delta_1, \delta_2)) \in VIEW(R^2|_{\gamma_1'',\gamma_2})(PK')$
- $(\hat{\beta}_2; (\gamma_1^1 \oplus \gamma_1', \gamma_2^1); (a', b_1', b_2', \beta_1', \beta_2', \delta_1', \delta_2')) \in VIEW(R^2|_{\gamma_1^1\oplus\gamma_1',\gamma_2^1})(PK')$
- $(\hat{\beta}_2; (\gamma_1^2, \gamma_2^2); (a'', b_1'', b_2'', \beta_1'', \beta_2'', \delta_1'', \delta_2'')) \in VIEW(R^2|_{\gamma_1^2,\gamma_2^2})(PK')$.

Equivalently, all the following conditions hold.

- 1. $\hat{\beta}_2 = comm_k(\beta_2; comm_k(a, b_1, b_2, \beta_1; r))$
  2. $(f_1^{UA}; \beta_1; \gamma_1'' \oplus \gamma_1'; \delta_1) \in VIEW(V^0(f, a, b_1))$
  3. $(f_2^{UA}; \beta_2; \gamma_2; \delta_2) \in VIEW(V^0(f, a, b_2))$.
- 1. $\hat{\beta}_2 = comm_k(\beta_2'; comm_k(a', b_1', b_2', \beta_1'; r))$
  2. $(f_1^{UA}; \beta_1'; (\gamma_1^1 \oplus \gamma_1') \oplus \gamma_1'; \delta_1') \in VIEW(V^0(f, a, b_1))$
  3. $(f_2^{UA}; \beta_2'; \gamma_2^1; \delta_2') \in VIEW(V^0(f, a, b_2))$.
- 1. $\hat{\beta}_2 = comm_k(\beta_2''; comm_k(a'', b_1'', b_2'', \beta_1''; r)$
  2. $(f_1^{UA}; \beta_1''; \gamma_1^2 \oplus \gamma_1'; \delta_1'') \in VIEW(V^0(f, a, b_1))$
  3. $(f_2^{UA}; \beta_2''; \gamma_2^2; \delta_2'') \in VIEW(V^0(f, a, b_2))$.

Since $comm_k$ is computationally-binding and $\tilde{S}_n$ is of polynomial-size, conditions (1) imply that

$$(a, b_1, b_2, \beta_1, \beta_2) = (a', b_1', b_2', \beta_1', \beta_2') = (a'', b_1'', b_2'', \beta_1'', \beta_2'').$$

The above equality combined with conditions (2) and (3) imply that

1. $(f_1^{UA}; \beta_1; \gamma_1^1; \delta_1') \in VIEW(V^0(f, a, b_1))$
2. $(f_2^{UA}; \beta_2; \gamma_2^2; \delta_2'') \in VIEW(V^0(f, a, b_2))$.

□

The proof-of-knowledge property of (P^0, V^0) implies that there exists a probabilistic-polynomial-time oracle machine E and a polynomial $p'(\cdot)$ such that for infinitely many n's, for $(a, b_1, b_2, aux_1, aux_2) = F_n(f, f_1^{UA}, f_2^{UA})$,

$$Pr\left[\begin{array}{c} \forall i \ E^{\tilde{P}_n(aux_1)}((f, a, b_1), i) = w_i^1 \text{ s.t. } ((f, a, b_1), w^1) \in \mathcal{R}_{\mathcal{F}} \\ and \\ \forall i \ E^{\tilde{P}_n(aux_2)}((f, a, b_2), i) = w_i^2 \text{ s.t. } ((f, a, b_2), w^2) \in \mathcal{R}_{\mathcal{F}} \end{array}\right] \geq \frac{1}{p'(n)}$$

(where the probability is over uniformly chosen $f, f_1^{UA}, f_2^{UA} \in \mathcal{F}_n$ and over the random coin tosses of F_n, $E^{\tilde{P}_n(aux_1)}$ and $E^{\tilde{P}_n(aux_2)}$).

Part 2: We next show how one can use E and F_n and $\tilde{P}_n$ to find a collision in $\mathcal{F}$. We define a probabilistic-polynomial-time oracle machine $\mathcal{M}$, which is given oracle access to E, F_n and $\tilde{P}_n$, and such that on input a random function $f \in \mathcal{F}_n$ outputs a collision in f, with non-negligible probability.

$\mathcal{M}^{E, F_n, \tilde{P}_n}$, on input $f \in \mathcal{F}_n$, operates as follows.

1. Choose $f_1^{UA}, f_2^{UA} \in_R \mathcal{F}_n$ and run $F_n(f, f_1^{UA}, f_2^{UA})$ to obtain $(a, b_1, b_2, aux_1, aux_2) \leftarrow F_n(f, f_1^{UA}, f_2^{UA})$.

2. Choose a random i, and compute $\hat{C_i^1}$ and $\hat{C_i^2}$ by emulating $E^{\tilde{P}_n(aux_1)}((f,a,b_1), 1+(i-1)((\lg n)^2+1))$ and $E^{\tilde{P}_n(aux_2)}((f,a,b_2), 1+(i-1)((\lg n)^2+1))$[17].

3. Compute the authentication path of $\hat{C_i^1}$ with respect to f, by emulating $E^{\tilde{P}_n(aux_1)}((f,a,b_1), 1+j+(i-1)((\lg n)^2+1))$ for $j = 1, ..., (\lg n)^2$.

4. Compute the authentication path of $\hat{C_i^2}$ with respect to f, by emulating $E^{\tilde{P}_n(aux_2)}((f,a,b_2), 1+j+(i-1)((\lg n)^2+1))$ for $j = 1, ..., (\lg n)^2$.

Claim B.0.4. *With non-negligible probability (over $f \in_R \mathcal{F}_n$ and over the random coin tosses of $\mathcal{M}$, E, F_n, and $\tilde{P}_n$) somewhere along these paths there will be a collision in f.*

Proof. With non-negligible probability (over the random coin tosses of $\mathcal{M}$, E, F_n, and $\tilde{P}_n$), $\hat{C_i^1} = E^{\tilde{P}_n(aux_1)}((f,a,b_1), 1+(i-1)((\lg n)^2+1))$ and $\hat{C_i^2} = E^{\tilde{P}_n(aux_2)}((f,a,b_2), 1+(i-1)((\lg n)^2+1))$, where $auth_f(\hat{C^1})$ is a witness of (f,a,b_1) in $\mathcal{R_F}$ and $auth_f(\hat{C^2})$ is a witness of (f,a,b_2) in $\mathcal{R_F}$. Also, with non-negligible probability, in steps 3 and 4 above E gives the authentication paths of $\hat{C_i^1}$ and $\hat{C_i^2}$.

Since $C_1 \neq C_2$ and since the circuit-encoding $C \to \hat{C}$ has large minimum distance, it follows that with probability $\frac{1}{poly}$, $\hat{C_i^1} \neq \hat{C_i^2}$ (where *poly* is a polynomial and the probability is over a randomly chosen i).

This implies that somewhere along these paths there will be a collision in f, since $\hat{C_i^1} \neq \hat{C_i^2}$ and yet $TC_f(\hat{C^1}) = TC_f(\hat{C^2}) = a$. Thus, we obtain a contradiction to our assumption that $\mathcal{F}$ is a collision resistant function ensemble. □

□

C Proof of Claim 6.5.1

Proof. Denote the output of $FORG^M(VK)$ by $(\hat{\beta}_2; (\gamma_1'', \gamma_2); (a, b_1, b_2, \beta_1, \beta_2, \delta_1, \delta_2))$.
By the definition of $\tilde{P}_2^n$, there exists a polynomial $p(\cdot)$ such that for every $n \in S^2_{\mathcal{H}^M}$ and for every $(a, b_1) = F_2^n$

$$Pr[(\tilde{P}_2^n, V^0)(f,a,b_1) = 1 \ \wedge \ (\tilde{P}_2^n, V^{\mathcal{H}^M})(f,a,b_2) = 1] \geq \frac{1}{p(n)} \qquad (1)$$

[17] Recall that we assumed that $\hat{C}_i$ is the k'th bit of the witness where $k = 1 + (i-1)((\lg n)^2+1)$.

(where the probability is over $f \in_R \mathcal{F}_n$, $b_2 \in_R \{0,1\}^n$ and the random coin tosses of V^0 and $V^{\mathcal{H}^M}$).
We claim that similarly, for every $n \in S^2_{\mathcal{H}^M}$ and for every $(a, b_1) = F_2^n$,

$$Pr[(\tilde{P}_2^n, V^0|_{f_1^{UA},\gamma_1''\oplus\gamma_1'})(f,a,b_1) = 1 \ \wedge \ (\tilde{P}_2^n, V^{\mathcal{H}^M}|_{q_M})(f,a,b_2) = 1] \geq \frac{1}{p(n)} \tag{2}$$

(where the probability is over $f \in_R \mathcal{F}_n$, $b_2 \in_R \{0,1\}^n$, and over f_1^{UA}, $\gamma_1'' \oplus \gamma_1'$ and q_M).
This is so for the following reasons

1. f_1^{UA} was chosen uniformly in $\mathcal{F}_n$
2. $\gamma_1'' \oplus \gamma_1'$ was chosen uniformly (follows from the fact that γ_1' was chosen uniformly and γ_1'' was chosen independently of γ_1').
3. $\tilde{P}_2^n$ (in step 7) cannot distinguish between the distribution of q_M and the distribution of a random query of $V^{\mathcal{H}^M}$.

For all of the above reasons, $\tilde{P}_n^2$ in (2) should succeed with essentially the same probability as in (1).

The fact that $(\tilde{P}_2^n, V^0|_{f_1^{UA},\gamma_1''\oplus\gamma_1'})(f,a,b_1) = 1$ implies that

- $(f_1^{UA}; \beta_1; \gamma_1'' \oplus \gamma_1'; \delta_1) \in VIEW(V^0(f,a,b_1))$.

The fact that $(\tilde{P}_2^n, V^{\mathcal{H}^M}|_{q_M})(f,a,b_2) = 1$ implies that $(q_M; ans) \in VIEW(V^{\mathcal{H}^M}(f,a,b_2))$, which in turn implies that both of the following conditions hold.

- $(f_2^{UA}; \beta_2; \gamma_2; \delta_2) \in VIEW(V^0(f,a,b_2))$
- $(\gamma_1'', \gamma_2) = h(\hat{\beta}_2, M))$.

The satisfaction of above three conditions imply that the forgery was successful. □

D Proof of Claim 6.7.1

Proof. We will prove that $Pr[(\tilde{\tilde{P}}_1^n, V^{\mathcal{G}})(f,a,b_1) = 1] \geq \frac{1}{p'(n)}$. The inequality, $Pr[(\tilde{\tilde{P}}_1^n, V^{\mathcal{H}^M})(f,a,b_2) = 1] \geq \frac{1}{p'(n)}$, can be proved in exactly the same manner. $\tilde{\tilde{P}}_1^n$, upon receiving a message $q = (f^{UA}, (k,r), g_1, \ldots, g_n)$ from $V^{\mathcal{G}}$, where $g_1, \ldots, g_n \in \mathcal{G}$, operates as follows.

1. Partition the g_i's into three sets S_1, S_2, S_3 of equal size.

2. Choose $h_1^M, \ldots, h_{\frac{n}{3}}^M \in_R \mathcal{H}^M$ and choose $h'_1, \ldots, h'_{\frac{n}{3}} \in_R \mathcal{H}'$.

3. Set

   - $q_1 = (f^{UA}, (k,r), S_1 \cup \{h_1^M, \ldots, h_{\frac{n}{3}}^M\} \cup \{h'_1, \ldots, h'_{\frac{n}{3}}\})$
   - $q_2 = (f^{UA}, (k,r), S_2 \cup \{h_1^M, \ldots, h_{\frac{n}{3}}^M\} \cup \{h'_1, \ldots, h'_{\frac{n}{3}}\})$
   - $q_3 = (f^{UA}, (k,r), S_3 \cup \{h_1^M, \ldots, h_{\frac{n}{3}}^M\} \cup \{h'_1, \ldots, h'_{\frac{n}{3}}\})$

4. Emulate $(\tilde{P}_1^n, V^{\mathcal{H}'}|q_i)(f,a,b_1)$, for $i = 1,2,3$, to obtain ans_1, ans_2, ans_3.

5. Output the parts of the answers of $\tilde{P}_1^n(f,a,b_1)$ that correspond to the g_i's. Namely, output $\{\beta_i, \hat{\beta}_i, \gamma_i, \delta_i\}_{i=1}^n$ where $\gamma_i = g_i(\hat{\beta}_i)$. and each $(\beta_i, \hat{\beta}_i, \gamma_i, \delta_i)$ is obtained from ans_1, ans_2 or ans_3.

The success of $\tilde{\tilde{P}}_1^n$ follows from the fact that q_1,q_2 and q_3 have the same distribution as a random q chosen by $V^{\mathcal{H}'}$. □

E Proof of Claim 6.7.2

Proof. Denote the output of the forger $FORG^M(VK)$ by $(\hat{\beta}_1, (\gamma_1, (b''_2, q'')), (a, b_1, \beta_1, \delta_1, ans))$. Claim 6.7.1 implies that there exists a poly-size circuit $\tilde{\tilde{P}}_1^n$ and a polynomial $p'(\cdot)$, such that for every $n \in S^1_{\mathcal{H}'}$ and for $a = F_1^n(f)$,

$$Pr[(\tilde{\tilde{P}}_1^n, V^{\mathcal{G}}(f,a,b_1) = 1 \ \wedge \ (\tilde{\tilde{P}}_1^n, V^{\mathcal{H}^M})(f,a,b_2) = 1] \geq \frac{1}{p'(n)} \tag{3}$$

(where the probability is over $f \in_R \mathcal{F}_n$, $b_1, b_2 \in_R \{0,1\}^n$ and the random coin tosses of $V^{\mathcal{H}^1}$ and $V^{\mathcal{H}^M}$).

We claim that similarly, for $a = F_1^n(f)$,

$$Pr[(\tilde{P}_1^n, V^{\mathcal{G}}|_{q' \oplus q''})(f,a,b_1) = 1 \ \wedge \ (\tilde{P}_1^n, V^{\mathcal{H}^M}|_{q_M})(f,a,b'_2 \oplus b''_2) = 1] \geq \frac{1}{p'(n)} \tag{4}$$

(where the probability is over $f \in_R \mathcal{F}_n,\ b_1, b'_2 \oplus b''_2 \in_R \{0,1\}^n$, $q' \oplus q''$, q_M).
This is so for the following reasons

1. $b'_2 \oplus b''_2$ is uniformly distributed in $\{0,1\}^n$.

2. $q' \oplus q''$ is uniformly distributed among the set of all queries of $V^{\mathcal{G}}$.

3. $\tilde{P}_1^n$ (in step $2(c)$) cannot distinguish between the distribution of q_M and the distribution of a uniform query of $V^{\mathcal{H}^M}$.

For all of the above reasons, $\tilde{P}_n^1$ in (4) should succeed with essentially the same probability as in (3).

Thus, for every $n \in S^1_{\mathcal{H}'}$, the following conditions hold with probability $\geq \frac{1}{p'(n)}$.

1. $(q' \oplus q''; ans) \in VIEW(V^{\mathcal{G}}(f, a, b'_2 \oplus b''_2))$.

2. $(q^M; ans^M) \in VIEW((V^{\mathcal{H}^M}(f, a, b_1))$, which in turn implies that the following conditions hold.

   (a) $\gamma_1 = h^M(\hat{\beta}_1)$, which implies that $(\gamma_1, (b''_2, q'')) = h(\hat{\beta}_1, M)$

   (b) $(f^{UA}; \beta_1; \gamma_1; \delta_1) \in VIEW(V^0(f, a, b_1))$

   (c) $\hat{\beta}_1 = comm_k(\beta_1; comm_k(a, b_1; r)$.

Recall that $VERIFY^3_{\mathcal{H}}(VK)$ accepts if conditions (1) and (2) hold, and thus for every $n \in S^1_{\mathcal{H}'}$, $FORG^M(VK)$ is successful with probability $\geq \frac{1}{p'(n)}$. □

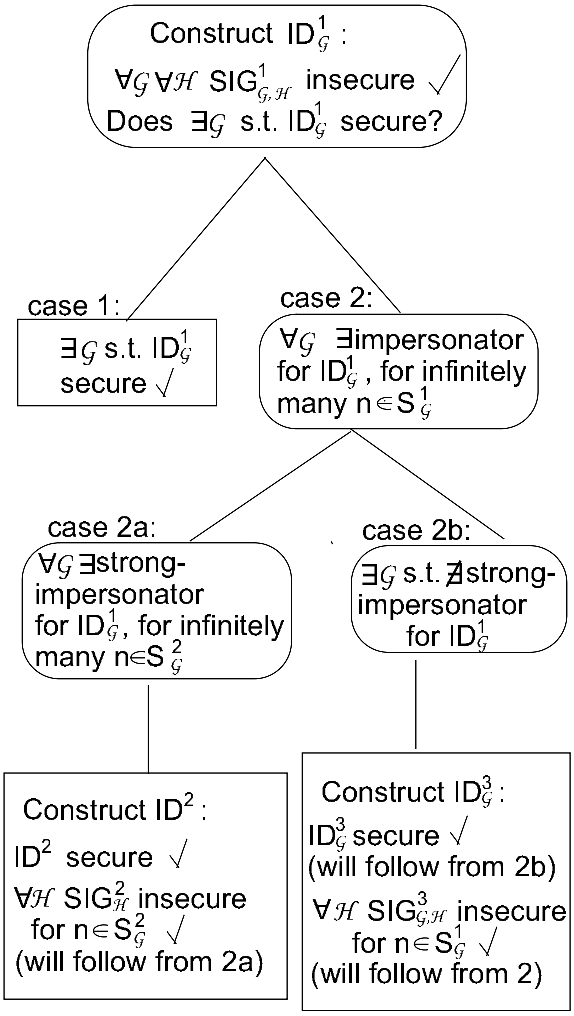
Construct ID$^1_{\mathcal{G}}$:
$\forall\mathcal{G}\forall\mathcal{H}$ SIG$^1_{\mathcal{G},\mathcal{H}}$ insecure $\surd$
Does $\exists\mathcal{G}$ s.t. ID$^1_{\mathcal{G}}$ secure?
case 1:
$\exists\mathcal{G}$ s.t. ID$^1_{\mathcal{G}}$ secure $\surd$
case 2:
$\forall\mathcal{G}$ $\exists$impersonator for ID$^1_{\mathcal{G}}$, for infinitely many n$\in$S$^1_{\mathcal{G}}$
case 2a:
$\forall\mathcal{G}$ $\exists$strong-impersonator for ID$^1_{\mathcal{G}}$, for infinitely many n$\in$S$^2_{\mathcal{G}}$
case 2b:
$\exists\mathcal{G}$ s.t. $\nexists$strong-impersonator for ID$^1_{\mathcal{G}}$
Construct ID^2 :
ID^2 secure $\surd$
$\forall\mathcal{H}$ SIG$^2_{\mathcal{H}}$ insecure for n$\in$S$^2_{\mathcal{G}}$ $\surd$
(will follow from 2a)
Construct ID$^3_{\mathcal{G}}$:
ID$^3_{\mathcal{G}}$ secure $\surd$
(will follow from 2b)
$\forall\mathcal{H}$ SIG$^3_{\mathcal{G},\mathcal{H}}$ insecure for n$\in$S$^1_{\mathcal{G}}$ $\surd$
(will follow from 2)

Figure 1: Proof of Theorem 2