Fiat-Shamir heuristics from crypto assumptions

by Tianqi Yang & Yizhi Huang

(Based on [Canetti-Chen-Holmgren-Lombardi-Rothblum-Rothblum-Wichs '19])

Last week: RSVL-hardness from the soundness of Fiat-Shamir heuristics.

Today:

- Summary of Fiat-Shamir and RSVL-hardness
- Fiat-Shamir heuristics in ROM
- Correlation intractable hash functions (CI)
- CI for functions from circular FHE [Canetti et. al. '19]

What Jiaqian talked about last week

Meta thm. For $\mathcal{L} \in$ PSPACE, $\mathcal{L}$ is hard + incrementally verifiable, unambiguous SNARG for $\mathcal{L}$ $\Longrightarrow$ RSVL hardness.

$\mathcal{L}$ = #SAT. P $\neq$ #P + Fiat-Shamir of sum-check protocol

(Choudhuri-Hubaćek-Kamath-Pietrzak-Rosen-Rothblum '19)

Follow-up works. How to construct Fiat-Shamir for $\mathcal{L}$.

① $\mathcal{L}$ = #SAT, instantiate FS for sumcheck. (#SAT hardness is implied by standard crypto assumptions)

Subexp LWE [Jawale-Kalai-Khurana-Zhang '21]

Subexp DDH [Kalai-Lombardi-Vaikuntanathan '22] (via [Jain-Jin '21])

② $\mathcal{L}$ = Iterated Squaring, instantiate FS for Pietrzak's protocol (Assuming IS is hard)

(Plain) LWE [Bitansky-Choudhuri-Holmgren-Kamath-Lombardi-Paneth-Rothblum '22]

improved upon [Lombardi-Vaikuntanathan '20]

Iterated Squaring

Given RSA group $\mathbb{Z}_n^\times$ where $n = pq$, $g \in \mathbb{Z}_n^\times$, t, compute $g^{2^t} \bmod n$.

(Straight-forward computation: $g \to g^2 \to g^{2^2} \to g^{2^3} \to \cdots \to g^{2^t}$, $O(t)$ time)

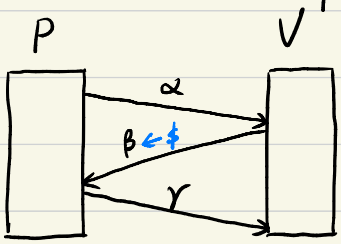
Today: The work that lies at the heart of all these works: [Canetti et. al. '19]

(with future works by [Peikert-Shiehian '19] & [Holmgren-Lombardi-Rothblum '21].)

Fiat-Shamir heuristics

(3-round)

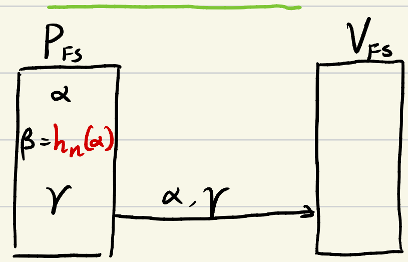
(Public-coin) interactive protocol



Fiat-Shamir $\Longrightarrow$

Non-interactive protocol (w/ CRS)

CRS: $\mathcal{H} = \{h_n\}$



Common Reference String (CRS): Sampled in advance, visible to everyone (P, V, adversary)

Introduced by Fiat & Shamir in 1986 to construct digital signature from identification schemes.

Why is FS useful: Simple & efficient

- Non-interactive zero-knowledge (NIZK)
- Succinct non-interactive argument (of knowledge) (SNARG/SNARK)
  ↳ related to PPAD hardness
- ...

In practice: Widely used w/ $\mathcal{H}$ = SHA256 (or other hash functions)

The worlds of protocols

	(Statistically-sound) proof	(Computationally sound) argument
Interactive	Interactive proof	Interactive argument
	↓ (Impossible)	↓ (Barriers known [Goldwasser-Kalai '03])
Non-interactive	Non-interactive proof	Non-interactive argument

Interactive proof → Non-interactive argument: Correlation intractable hash

Remark [Bitansky-Dachman-Soled-Garg-Jain-Kalai-López-Alt-Wichs '13]

General FS does not follow from falsifiable assumptions.

Warm-up. FS in random oracle model (ROM)

Thm. Taking $\mathcal{H}$ as random oracle can securely instantiate FS (from argument to argument)

Proof. We need to show that

if there is a p.p.t. cheating prover P^*_{FS} that makes the V^*_{FS} after FS accepts w/ non-negl. prob.

then we can construct a p.p.t. cheating prover P^* that breaks V before FS.

Intuition. P^*_{FS} will have to make the query α to the RO.

(O.w. P^*_{FS} can fool V_{FS} with prob. at most $2^{-|\beta|}$

Our P^* basically simulates P^*_{FS}.

Suppose P^*_{FS} makes T queries to the RO (Suppose that the queries are non-adaptive)

P^*: Randomly sample an index $i \leftarrow [T]$.

Answer all queries by P^*_{FS} except the i^{th} one by random string.

Suppose the i^{th} query of P^*_{FS} is α, send α to V in the first round,

and treat the response β by V as the oracle output to α.

Complete the rest of the protocol, suppose the last message sent by P^*_{FS} is (α, γ),

send γ as the third message.

W.p. $\frac{1}{T} = \frac{1}{\text{poly}}$, P_{FS} guesses i correctly.

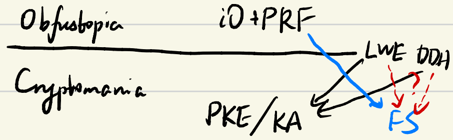
In such case, P^*_{FS} breaks $V_{FS} \Rightarrow P^*$ breaks V. $\boxed{\text{Z}}$

Remark. $i\mathcal{O}$ + PRF can be viewed as RO informally.

Cor. subexp $i\mathcal{O}$ + subexp PRF + ⋯ $\Rightarrow$ Fiat-Shamir $\Rightarrow$ RSVL hardness. [Kalai-Rothblum-Rothblum'17]

(reproving a weaker result presented two weeks ago)

Q: Can we instantiate FS from weaker crypto assumptions?



Correlation intractable hash function (CI hash)

Relation. $R \subseteq \mathcal{X} \times \mathcal{Y}$.

Def (CI hash) Let $S = \{S_n\}_{n \geq 0}$ be a class of relations. A hash family $\mathcal{H} = \{h_n(k, x)\}$ is called correlation intractable for S if for any p.p.t. adversary $\mathcal{A}$ and relation R,

$$\Pr\left[\begin{array}{l} k \leftarrow \$ \\ x \leftarrow \mathcal{A}(k) \end{array} ; \ (x, h_n(k, x)) \in R\right] < \text{negl}.$$

We call S the bad relations.

Intuition. A hash is CI if any p.p.t. adv. cannot find x w/ $(x, h(x))$ being bad.

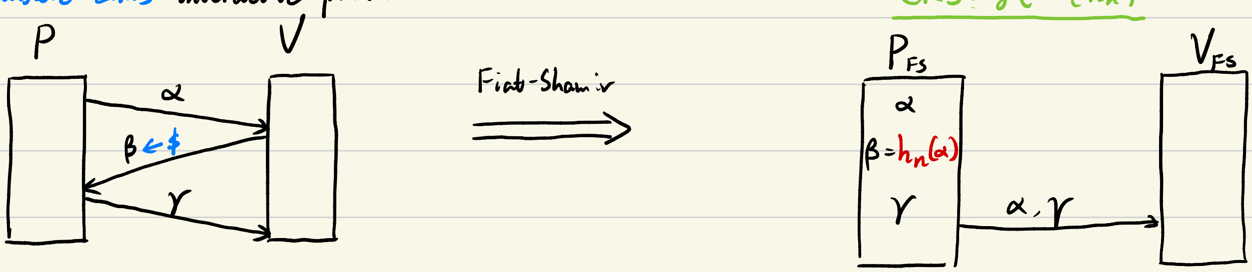
Def. A relation is called sparse if $\forall x$, $\Pr_y[(x, y) \in R] < \text{negl}$.

Claim. Any CI hash for the class of all sparse relations securely instantiates FS.

Proof. (3-round) (Public-coin) interactive protocol

Non-interactive protocol (w/ CRS)

CRS: $\mathcal{H} = \{h_n\}$



Bad relation. $(\alpha, \beta) \in R$ iff $\exists \gamma$ s.t. V accepts.

(P, V) is sound $\Rightarrow \Pr_\beta[(\alpha, \beta) \in R] < \text{negl}$. Hence R is sparse.

So CI guarantees that it is hard to find such a "bad" α s.t. $(\alpha, h_n(\alpha)) \in R$.

So (P_{FS}, V_{FS}) is sound. $\boxtimes$

CI constructions

(For sparse relations) [Canetti-Chen-Reyzin-Rothblum'18] Strong KDM-secure encryption w/ universal ciphertexts.

(For functions in P) $\left\{ \begin{array}{l} \text{[Canetti et al. '19] Circular FHE.} \\ \text{[Peikert-Shiehian '19] LWE} \\ \text{[Jain-Jin '21] Subexp DDH} \end{array} \right.$

$\forall x$ there is at most one y s.t. $(x,y) \in R$.

CI hash for functions from circular FHE [Canetti et al '19]

Fully homomorphic encryption (FHE)

FHE is a public-key encryption scheme (Gen, Enc, Dec) s.t.

(Correctness) $(pk, sk) \leftarrow Gen(1^\lambda)$, $ct \leftarrow Enc(pk, m; r) \Longrightarrow m = Dec(sk, ct)$

(Security) $\{(pk, sk) \leftarrow Gen(1^\lambda), (pk, Enc(pk, 0))\} \approx_c \{(pk, sk) \leftarrow Gen(1^\lambda), (pk, Enc(pk, 1))\}$.

w/ an additional Eval that evaluates any circuit $C: \{0,1\}^n \to \{0,1\}$ on the ciphertext.

(Fully-homomorphism) $Dec(sk, Eval(pk, C, Enc(pk, m_1), Enc(pk, m_2), \cdots, Enc(pk, m_n))) = C(m_1, \cdots, m_n)$.

Circular security $\{(pk, sk) \leftarrow Gen(1^\lambda), (pk, Enc(pk, 0))\} \approx_c \{(pk, sk) \leftarrow Gen(1^\lambda), (pk, Enc(pk, sk))\}$

(Encryption of messages depending on sk is still secure)

Def. A relation R is called function if there is $f: \mathcal{X} \to \mathcal{Y} \cup \{\perp\}$ s.t. $(x, y) \in R$ iff $y = f(x)$.

We say it is computable in time T if f is computable in time T.

CI for function $\Pr[k \leftarrow \$, x \leftarrow \mathcal{A}(k) : h(x) = f(x)] < negl$.

Thm. If circular FHE exists, then for any $c > 0$, let S_c be the class of all functions computable in n^c time, there is a CI hash for S_c.

Intuition. If S only contains one function f, then $h(x) \triangleq f(x) \oplus 1$ is a CI hash for f.

Proof. Fix $c > 0$, let $c' > 0$ be a sufficiently large constant.

Construction. $(pk, sk) \leftarrow Gen(1^\lambda)$

$\hat{ct} \leftarrow Enc(pk, 0^{n^{c'}})$

Hash key $k \triangleq (pk, \hat{ct})$

$\Downarrow$

$h(k, x) \triangleq Eval(pk, \mathcal{U}_x, \hat{ct})$ where $\mathcal{U}_x(C) \triangleq C(x)$ for an encoding of circuit C.

Proof of security. Observation. $\hat{ct} \leftarrow Enc(pk, 0^{n^{c'}})$ is indistinguishable from $Enc(pk, g(\cdot))$ for any function g of description length $\leq n^{c'}$, by security.

So we can replace the $\hat{ct}$ in the def of hash by any $Enc(pk, g(\cdot))$.

Plug in $g(x) \triangleq Dec(sk, f(x)) \oplus 1$. $g(x)$ depends on $sk \Rightarrow$ circular security!

Suppose this is not a CI hash, then there is an adversary finding x s.t. $h(k, x) = f(x)$ w/ non-negl. prob.

Here comes the magic:

$$
\begin{aligned}
f(x) &= h(k, x) \\
&= Eval(pk, \mathcal{U}_x, Enc(pk, \langle Dec(sk, f(\cdot)) \oplus 1 \rangle))
\end{aligned}
$$

Apply Dec to both sides:

$$
\begin{aligned}
Dec(sk, f(x)) &= \mathcal{U}_x(\langle Dec(sk, f(\cdot)) \oplus 1 \rangle) \\
&= Dec(sk, f(x)) \oplus 1.
\end{aligned}
$$

Contradiction! $\boxed{\lightning}$