Provable security models for distance-bounding

Ioana Boureanu

UNIVERSITY OF SURREY

http://people.itcarlson.com/ioana/

based on joint works with with S. Vaudenay, K. Mitrokotsa & discussions with D. Gerault, G. Avoine and quite a few others

How NOT to Analyse Security Against PPT Attackers?

- we should not play a “cat-and-mouse” game

fix **new** attack

Cryptoscheme 1 — secure against attack 1

Cryptoscheme 2 — secure against **new** attack

...

pic: S. Faust

- what we’d *really* need to show is security NOT against one attack but against a broad range of attackers

How to Analyse Security Against PPT Attackers?

1. Give a **security definition**

   What is the security property that the scheme should achieve?

2. Define **attacker model**

   How can the attacker interact with the scheme?

3. If needed, make an **assumption**

   What do you pre-suppose for the security to hold?

4. Do the **proof**

   Prove that scheme satisfies the security definition, if assumption holds

   $\Rightarrow$ the only way to break the scheme is to break assumption

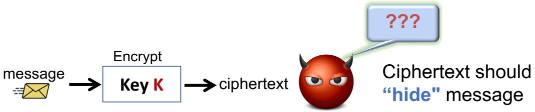
   - $\Rightarrow$ Secure against **any** attack
   - **any** attack within the model that does not break the assumption

5. Verify **proof**

How to Analyse Security Against PPT Attackers?

1. Give a **security definition**

   What is the security property that the scheme should achieve?



2. Define **attacker model**

   How can the attacker interact with the scheme?

3. If needed, make an **assumption**

   What do you pre-suppose for the security to hold?

4. Do the **proof**

   Prove that scheme satisfies the security definition, if assumption holds

   $\Rightarrow$ the only way to break the scheme is to break assumption

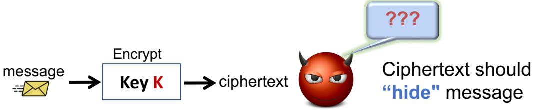
   - $\Rightarrow$ Secure against **any** attack
   - **any** attack within the model that does not break the assumption

5. Verify **proof**

How to Analyse Security Against PPT Attackers?

1. Give a **security definition**

   What is the security property that the scheme should achieve?



2. Define **attacker model**

   How can the attacker interact with the scheme?

3. If needed, make an **assumption**

   What do you pre-suppose for the security to hold?

4. Do the **proof**

   Prove that scheme satisfies the security definition, if assumption holds

   $\Rightarrow$ the only way to break the scheme is to break assumption

   - $\Rightarrow$ Secure against **any** attack
   - **any** attack within the model that does not break the assumption

5. Verify **proof**

Why Security Definitions?

We need to know **what we want** in order to achieve it

Why definitions?

Allows to **compare schemes**: some definitions may be stronger than others

Allows for **proofs**: security proof only meaningful with definition

Coming up with the right definition is non-trivial

Examples in public-key encryption, TLS, etc.

What About The Rest... Besides Definitions?

- for meaningful provable security, we need
  - the attacker model be suited to the application (debatable)
  - the proof be correct (NOT debatable)
- these are also non-trivial & often hard to argue and, respectively check

Why Provable Security for DB? (I)

- we've played the "cat-and-mouse" game 
  - many arguments along the best-attack scenarios ...
  - many insecurities proven ...

– **in a model without communication noise**, best-known **symmetric-key** DB protocols and success probabilities of best-known attacks ($\theta < 1$ constant s.t. $2^{-\theta n}$ negligible), by 2015

	Protocol	Success Probability		
		Distance-Fraud	MiM	Terrorist-Fraud
†	**Brands & Chaum**	$(1/2)^n$	$(1/2)^n$	$1, \text{negl}$
†	**Bussard & Bagga**	1	$(1/2)^n$	$1, \text{negl}$
†	**Čapkun *et al.***	$(1/2)^n$	$(1/2)^n$	$1, \text{negl}$
†	**Hancke & Kuhn**	$(3/4)^n$–1	$(3/4)^n$	$1, \text{negl}$
†	**Reid *et al.***	$(3/4)^n$–1	1	$(3/4)^{\theta n}, \text{negl}$
†	**Singelée & Preneel**	$(1/2)^n$	$(1/2)^n$	$1, \text{negl}$
†	**Tu & Piramuthu**	$(3/4)^n$	1	$(3/4)^{\theta n}, \text{negl}$
†	**Munilla & Peinado**	$(3/4)^n$	$(3/5)^n$	$1, \text{negl}$
☹	**Swiss-Knife**	$(3/4)^n$	$(1/2)^n$–1	$(3/4)^{\theta n}, \text{negl}$
†	**Kim & Avoine**	$(7/8)^n$	$(1/2)^n$	$1, \text{negl}$
†	**Nikov & Vauclair**	$1/k$	$(1/2)^n$	$1, \text{negl}$
☹	**Avoine *et al.***	$(3/4)^n$–1	$(2/3)^n$–1	$(2/3)^{\theta n}, \text{negl}$
☺	**SKI**	$(3/4)^n$	$(2/3)^n$	γ, γ'
☺	**Fischlin & Onete**	$(3/4)^n$	$(3/4)^n$	$\gamma = \gamma'$
☺	**DB1**	$(1/2)^n$	$(1/3)^n$	$(2/3)^{\theta n}$
☺	**DB2**	$(1/\sqrt{2})^n$	$(1/2)^n$	$(1/\sqrt{2})^{\theta n}$

Why Provable Security for DB? (II)

- so, we've played the "cat-and-mouse" game 
- many incorrect arguments for DB, in some existing proofs (e.g., insufficient assumptions or used assumptions wrongly, etc.)

Why Provable Security for DB? (II)

- so, we've played the "cat-and-mouse" game 
- many incorrect arguments for DB, in some existing proofs (e.g., insufficient assumptions or used assumptions wrongly, etc.)

Security Proofs Based on PRF

Recall assumptions...

• *the PRF assumption*: for a set of functions of arbitrary-length input and arbitrary-length output indexed on a set of keys which is a PRF, no ppt can distinguish an instance from the PRF family (sampled uniformly) from a real random function over the same domains

- if the adversary can break the scheme with a PRF, then he can break an idealised scheme whereby the PRF is replaced by a truly random function
- this argument is valid when both conditions below are met:
  - the adversary does not have access to the PRF key
  - the PRF key is only used by the PRF
- as far as distance-fraud is concerned, condition 1 is not met!
- in many designs for terrorist-fraud resistance, condition 2 is not met!

Security Proofs Based on PRF

Recall assumptions...

• *the PRF assumption*: for a set of functions of arbitrary-length input and arbitrary-length output indexed on a set of keys which is a PRF, no ppt can distinguish an instance from the PRF family (sampled uniformly) from a real random function over the same domains

- if the adversary can break the scheme with a PRF, then he can break an idealised scheme whereby the PRF is replaced by a truly random function
- this argument is valid when both conditions below are met:
  - the adversary does not have access to the PRF key
  - the PRF key is only used by the PRF
- as far as distance-fraud is concerned, condition 1 is not met!
- in many designs for terrorist-fraud resistance, condition 2 is not met!

Security Proofs Based on PRF

Recall assumptions...

• *the PRF assumption*: for a set of functions of arbitrary-length input and arbitrary-length output indexed on a set of keys which is a PRF, no ppt can distinguish an instance from the PRF family (sampled uniformly) from a real random function over the same domains

- if the adversary can break the scheme with a PRF, then he can break an idealised scheme whereby the PRF is replaced by a truly random function
- this argument is valid when both conditions below are met:
  - the adversary does not have access to the PRF key
  - the PRF key is only used by the PRF
- as far as distance-fraud is concerned, condition 1 is not met!
- in many designs for terrorist-fraud resistance, condition 2 is not met!

Security Proofs Based on PRF

Recall assumptions...

• *the PRF assumption*: for a set of functions of arbitrary-length input and arbitrary-length output indexed on a set of keys which is a PRF, no ppt can distinguish an instance from the PRF family (sampled uniformly) from a real random function over the same domains

- if the adversary can break the scheme with a PRF, then he can break an idealised scheme whereby the PRF is replaced by a truly random function
- this argument is valid when both conditions below are met:
  - the adversary does not have access to the PRF key
  - the PRF key is only used by the PRF
- as far as distance-fraud is concerned, condition 1 is not met!
- in many designs for terrorist-fraud resistance, condition 2 is not met!

Why Provable Security for DB? (III)

- so, we've played the "cat-and-mouse" game 
- so, many incorrect arguments for DB in some existing proofs (e.g., PRF assumption used wrongly.)

  - [Boureanu-Mitrokotsa-Vaudenay Latincrypt 2012]:
    many DF attacks and MiM attacks,
    by "programmable PRFs"
    in protocols where the security claim was
    "if the PRF assumption holds, then the protocol is secure"

  - the PRF assumption holding may not always be sufficient an assumption for DB security !

  - design solutions/correction put in place, PRF masking, circular-keying PRF security, but they needed bringing together [Boureanu Mitrokotsa Vaudenay 2013 – 2015]

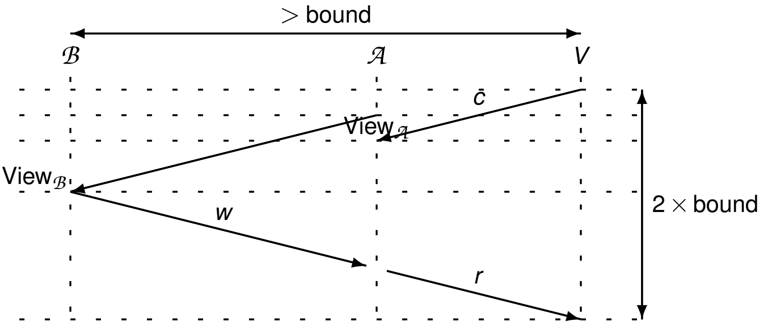
The “BMV” Model: The Beginnings...

- appeared in 2013 [Boureanu-Mitrokotsa-Vaudenay ISC 2013]
- continued to evolve ...[Boureanu-Vaudenay Inscrypt 2014], [Boureanu-Mitrokotsa-Vaudenay JoCS 2015]... and beyond
- the “BMV” model is based on the principle of interactive proofs
- a formal model, dubbed “DFKO”, existed from before [Dürholz-Fischlin-Kasper-Onete ISC 2011]; session-based (different “patterns” over sessions to model relaying and MiM.. and TF)
- a formal framework, existed from before [Avoine et al 2009], which is more an attacker-model framework than a full provable-security formalism

BMV: Explicit Time in the Communication Model

- participants have **location**
- insecure broadcasting + messages have a purported destination
- all communication are subject to a **transmission-speed** limit
- a message sent at time t_{sent} from loc_A is visible at loc_B at time $t_{received} \geq t_{sent} + d(loc_A, loc_B)$
- several adversarial instances, each with a location
- multiple instances but one **distinguished instance of** V; instances within a distance $\leq B$ are close-by; others are far-away
- adversaries can impersonate and change the message destination but cannot defeat the laws of physics: a malicious instance at loc_M, at time t_{act} could to block messages from loc_A to loc_B received at time $t_{received} \geq t_{act} + d(loc_M, loc_B)$
- honest instances only see messages for which they are purported recipient
- all communication is subject to random noise
  - adversaries receive noiseless communication
  - when time is not considered, honest participants receive noiseless messages

BMV: "Fundamental" Lemma — used in security proofs



Lemma

For each U, let View$_U$ *be his view just before receiving c. We say that a message by U is independent from c if it is the result of applying U on* View$_U$*, or a prefix of it.*

There exists A and a list w of messages independent from c such that if V receives r within at most $2 \times$ bound *time, then* $r = A(\text{View}_{\mathcal{A}}, c, w)$.

BMV: DB Experiment as Interactive Proofs – Summary

- **interactive proof** for proximity [Boureanu-Vaudenay Inscrypt 2014]
  - – a verifier party (its instances are honest)
  - – a prover party (its instances may be malicious)
  - – a secret to characterise the prover (in the symmetric case)
  - – concurrency: many provers+ verifiers + malicious participants
- **correctness/completeness**:
  - – if the honest prover is close to the verifier, the verifier accepts
- **"honest-prover" security**:
  - • Pr[V accepts]= *negl*, for any experiment where:
    - – the prover is honest and
    - – all its instances are far-away
  - • captures man-in-the-middle, impersonation, relay attack, mafia-fraud
- **soundness**:
  - • a honest prover does not leak (too much) secret information
  - • captures terrorist-fraud
- **(generalised) distance-fraud resistance** (capturing distance-hijacking)
- **distance-hijacking resistance** [Vaudenay FC 2015]

BMV: A Glance at a Generalised DB Threat Model

(Generalised) Distance-Fraud

Definition (α-resistance to distance-fraud). $(\forall s)\ (\forall P^*)\ (\forall loc_V \text{ such that } d(loc_V, loc_{P^*}) > \mathbb{B})\ (\forall r_k)$, *we have*

$$\Pr_{r_V}\left[Out_V = 1 : \begin{array}{l}(x,y) \leftarrow Gen(1^s; r_k) \\ P^*(x) \longleftrightarrow V(y; r_V)\end{array}\right] \leq \alpha$$

where P^* is any (unbounded) dishonest prover. In a concurrent setting, we implicitly allow a polynomially bounded number of honest $P(x')$ and $V(y')$ close to $V(y)$ with independent (x', y').

- **generalised distance fraud**:
  - $P(x)$ far from all $V(x)$'s want to make one $V(x)$ accept (interaction with other $P(x')$ and $V(x')$ possible anywhere)
  - $\rightarrow$ also captures distance hijacking
- generalised **mafia fraud**, to **MiM**:
  - *learning phase*: $\mathcal{A}$ interacts with many P's and V's
  - *attack phase*: $P(x)$'s far away from $V(x)$'s, $\mathcal{A}$ interacts with them and possible $P(x')$'s and $V(x')$'s, $\mathcal{A}$ wants to make one $V(x)$ accept
- generalised **terrorist fraud**, to **collusion fraud**:
  - $P(x)$ far from all $V(x)$'s interacts with $\mathcal{A}$ and makes one $V(x)$ accept, but View$(\mathcal{A})$ does not give any advantage to mount a man-in-the-middle attack

Man-in-the-Middle: More details

Practical & Provably Secure Distance-Bounding [Boureanu-Mitrokotsa-Vaudenay ISC 2013, JoCS 2015]

(Generalised) Mafia-Fraud

Definition (β-resistance to MiM). $(\forall s)(\forall m,\ell,z)$ *polynomially bounded,* $(\forall \mathcal{A}_1,\mathcal{A}_2)$ *polynomially bounded, for all locations such that* $d(loc_{P_j}, loc_V) > \mathbb{B}$*, where* $j \in \{m+1,\ldots,\ell\}$*, we have*

$$\Pr\left[Out_V = 1 : \begin{array}{c} (x,y) \leftarrow Gen(1^s) \\ P_1(x),\ldots,P_m(x) \longleftrightarrow \mathcal{A}_1 \longleftrightarrow V_1(y),\ldots,V_z(y) \\ P_{m+1}(x),\ldots,P_\ell(x) \longleftrightarrow \mathcal{A}_2(View_{\mathcal{A}_1}) \longleftrightarrow V(y) \end{array} \right] \leq \beta$$

over all random coins, where $View_{\mathcal{A}_1}$ *is the final view of* $\mathcal{A}_1$*. In a concurrent setting, we implicitly allow a polynomially bounded number of* $P(x')$, $P^*(x')$*, and* $V(y')$ *with independent* (x',y')*, anywhere.*

Definition (β-resistance to Man-in-the-Middle)

We say that a DB protocol is β-*resistant to MiM* if

$$\forall \mathcal{A}_1, \mathcal{A}_2 \quad \Pr\left[V_2 \text{ accepts} : \begin{array}{c} \overbrace{P_1 \longleftrightarrow \mathcal{A}_1 \longleftrightarrow V_1}^{\text{no restriction}} \\ \underbrace{P_2 \longleftrightarrow \mathcal{A}_2(\text{View}(\mathcal{A}_1)) \longleftrightarrow V_2}_{\text{far away}} \end{array} \right] \leq \beta$$

P_2 and V_2 are far away

- captures relay attacks; man-in-the-middle attacks; impersonation; leakage of credentials

Recall: Terrorist-fraud (TF) + TF-resistance

$$\underbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}_{\text{far away}}$$

- **informally, valid TF**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier V, without giving $\mathcal{A}$ another advantage ("advantage" often equates to key-leakage)

- **TF-resistance**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier $V \Rightarrow \mathcal{A}$ gets an advantage, i.e.,

  $$\forall P^*, \mathcal{A}.\;\; \Pr[\, V \text{ accepts}\,] \text{ high } \Rightarrow \exists B.\; \Pr[\, B(\mathit{ViewA}) \text{ passes}\,] \text{ high}$$

- **formally show a TF**: exhibit some $(P^*, \mathcal{A})$ such that $\Pr[\, V \text{ accepts}\,]$ high , and then show that $\forall B$, $\Pr[\, B(\mathit{ViewA}) \text{ passes}\,]$ negl

- .... unusual security property

Recall: Terrorist-fraud (TF) + TF-resistance

$$\underbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}_{\text{far away}}$$

- **informally, valid TF**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier V, without giving $\mathcal{A}$ another advantage ("advantage" often equates to key-leakage)

- **TF-resistance**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier $V \Rightarrow \mathcal{A}$ gets an advantage, i.e.,

  $$\forall P^*, \mathcal{A}.\ \ \Pr[\,V \text{ accepts}\,] \text{ high } \Rightarrow \exists B.\ \Pr[\,B(\mathit{ViewA}) \text{ passes}\,] \text{ high}$$

- **formally show a TF**: exhibit some $(P^*, \mathcal{A})$ such that $\Pr[\,V \text{ accepts}\,]$ high , and then show that $\forall B$, $\Pr[\,B(\mathit{ViewA}) \text{ passes}\,]$ negl

- .... unusual security property

Recall: Terrorist-fraud (TF) + TF-resistance

$$\underbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}_{\text{far away}}$$

- **informally, valid TF**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier V, without giving $\mathcal{A}$ another advantage ("advantage" often equates to key-leakage)

- **TF-resistance**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier $V \Rightarrow \mathcal{A}$ gets an advantage, i.e.,

  $$\forall P^*, \mathcal{A}.\;\; \Pr[V \text{ accepts }] \text{ high } \Rightarrow \exists B.\; \Pr[B(\mathit{ViewA}) \text{ passes }] \text{ high}$$

- **formally show a TF**: exhibit some $(P^*, \mathcal{A})$ such that $\Pr[V \text{ accepts }]$ high , and then show that $\forall B$, $\Pr[B(\mathit{ViewA}) \text{ passes }]$ negl

- .... unusual security property

Recall: Terrorist-fraud (TF) + TF-resistance

$$\underbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}_{\text{far away}}$$

- **informally, valid TF**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier V, without giving $\mathcal{A}$ another advantage ("advantage" often equates to key-leakage)

- **TF-resistance**: a malicious, far-away prover P^* helps an adversary $\mathcal{A}$ to show that P^* is close to a verifier $V \Rightarrow \mathcal{A}$ gets an advantage, i.e.,

  $$\forall P^*, \mathcal{A}.\ \ \Pr[V \text{ accepts }] \text{ high } \Rightarrow \exists B.\ \Pr[B(\mathit{ViewA}) \text{ passes }] \text{ high}$$

- **formally show a TF**: exhibit some $(P^*, \mathcal{A})$ such that $\Pr[V \text{ accepts }]$ high , and then show that $\forall B$, $\Pr[B(\mathit{ViewA}) \text{ passes }]$ negl

- .... unusual security property

"DFKO": SimTF Definition for TF-resistance ...

[Düerholz-Fischlin-Kasper-Onete ISC2011]

SimTF Security

We say that a DB protocol is *SimTF-secure* if

$$\forall P^*, \mathcal{A},\ \exists B \text{ s. that } p_B \geq p_A,$$

where

$$p_A = \Pr[V \text{ accepts in } P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V\ ]$$

$$p_B = \Pr[V \text{ accepts in } B(View(\mathcal{A})) \longleftrightarrow V\ ].$$

and

in $P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V$ there is NO adversarial interaction in the rapid-bit exchange phase

... Hmmm, but OK ...

“BMV”: TF-resistance v0.1

[Boureanu-Mitrokotsa-Vaudenay Lightsec 2013]

(γ, γ')-resistance to TF

We say that a DB protocol is (γ, γ')-*resistance to TF* if

$$\forall P^*, \mathcal{A}, \ \exists B \text{ s. that } p_A \geq \gamma \Rightarrow p_B \geq \gamma'$$

where P^* and V are far-away and

$$p_A = \Pr[V \text{ accepts in } P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V\ ]$$

$$p_B = \Pr[V \text{ accepts in } B(View(\mathcal{A})) \longleftrightarrow V\ ].$$

“BMV vs FO”: SimTF vs. (γ, γ')-resistance to TF

Modulo Some Difference in what $\mathcal{A}$ can do...

SimTF-secure $\Leftrightarrow (\gamma, \gamma)$-resistant to TF

"BMV": Collusion-Fraud Resistance v1

[Boureanu-Mitrokotsa-Vaudenay ISC 2013]

collusion-fraud, informally: $P(x)$ far from all $V(x)$, interacts with $\mathcal{A}$ and $V(x)$ accepts on this, but $View(\mathcal{A})$ does any not give $\mathcal{A}$ any further advantage to mount a MiM

Definition ((γ,γ')-resistance to CF)

We say that a DB protocol is (γ,γ')-*resistant to CF* if

$$\forall P^*, \mathcal{A}\ \exists \mathcal{B}\ \ p_{\mathcal{A}} \geq \gamma \Longrightarrow p_{\mathcal{B}} \geq \gamma'$$

$$p_{\mathcal{A}} = \Pr\left[V \text{ accepts} : \overbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}^{\text{far away}} \right]$$

$$p_{\mathcal{B}} = \Pr\left[V \text{ accepts} : \underbrace{P \longleftrightarrow \mathcal{B}(\mathrm{View}(\mathcal{A})) \longleftrightarrow V}_{\text{far away}} \right]$$

“DFKO”: Game-TF Security

[Fischlin-Onete ACNS 2013]

Definition (γ-GameTF security)

We say that a DB protocol is γ-*GameTF-secure* if

$$\forall P^*, \mathcal{A}\ \exists \mathcal{B}\ \ p_{\mathcal{A}} \geq \gamma \Longrightarrow p_{\mathcal{B}} \geq \mathrm{Adv}^{\mathrm{MF}}$$

$$p_{\mathcal{A}} = \Pr\left[V \text{ accepts} : \overbrace{P^* \longleftrightarrow \mathcal{A} \longleftrightarrow V}^{\text{far away}} \right]$$

$$p_{\mathcal{B}} = \Pr\left[V \text{ accepts} : \underbrace{P \longleftrightarrow \mathcal{B}(\mathrm{View}(\mathcal{A})) \longleftrightarrow V}_{\text{far away}} \right]$$

$\mathrm{Adv}^{\mathrm{MF}}$ is the best probability that a verifier accepts in a mafia-fraud attack. (For adversaries with bounded complexity.)

“BMV vs FO”: GameTF vs. (γ, γ')-resistance to CF

Modulo Some Difference in what $\mathcal{A}$ can do...

GameTF-secure $\Leftrightarrow (\gamma, Adv^{MF})$-resistant to CF

“BMV”: Soundness v1 ...

[Vaudenay ProvSec 2013] and [Boureanu-Vaudenay Inscrypt 2014]

IDEA:

– if the verifier accepts with probability at least γ, then one can extract the secret from the view of close-by participants (which here is $\mathcal{A}$)

Definition **((γ, γ', m)-soundness).** *We say that a DB protocol is* <u>(γ, γ', m)-sound</u> *if for any distinguished experiment* $\mathsf{exp}(\mathcal{V})$ *in which* $\mathcal{V}$ *accepts with probability at least* γ*, there exists a PPT algorithm* $\mathcal{E}$ *called* extractor*, with the following property. By* $\mathcal{E}$ *running experiment* $\mathsf{exp}(\mathcal{V})$ *several times, in some executions denoted* $\mathsf{exp}_i(\mathcal{V})$*,* $i = 1, \ldots, M$*, for* M *of expected value bounded by* m*, we have that*

$$\Pr\left[\mathsf{Out}_V = 1 : \mathcal{E}(\mathsf{View}_1, \ldots, \mathsf{View}_M) \leftrightarrow \mathcal{V} \middle| \mathsf{Succ}_1, \ldots, \mathsf{Succ}_M\right] \geq \gamma',$$

where View_i *denotes the view of all close-by participants (except* $\mathcal{V}$*) and the transcript seen by* $\mathcal{V}$ *in the run* $\mathsf{exp}_i(\mathcal{V})$*, and* Succ_i *is the event that* $\mathcal{V}$ *accepts in the run* $\mathsf{exp}_i(\mathcal{V})$.

Definition (γ-m-soundness)

We say that a DB protocol is *γ-m-sound* if

$$\forall \mathsf{exp}\ \exists \mathcal{E}$$
$$\Pr[\mathcal{E}(\mathsf{View}_1, \ldots, \mathsf{View}_m) = x | \mathsf{Succ}_1, \ldots, \mathsf{Succ}_m] = 1 - \mathsf{negl}(n)$$

exp is an experiment such that

- provers are far away
- $\Pr[V \text{ accepts}] \geq \gamma$

$\mathcal{E}$ runs m times exp: $\mathsf{exp}_1, \ldots, \mathsf{exp}_m$,
View_i denotes the view of all close-by participants in exp_i
Succ_i is the event that V accepts in exp_i

"BMV": γ-m-soundness vs. (γ, γ')-resistance to CF

Theorem

$$\gamma\text{-}m\text{-soundness} \Rightarrow (\gamma, 1 - \mathrm{negl})\text{-resistance to CF},$$

for γ such that γ^{-1} is polynomially bounded

Protocols and Proofs...

... in “BMV” model

- Handan Kilinc – tomorrow

... in “DFKO” model

- David Gerault – tomorrow

Challenges

- the BMV and FO models do have difference in time-modelling, in relay-modelling, in what the attackers can do ...
- plenty of security definitions (too many?) to suit different designs? (OK or KO?)
- these definitions do NOT always overlap (especially if we do not iron out model-differences)
- TF-resistance hinders designs (i.e., renders them communication-expensive), yields hard-to-follow proofs, generally lowers MiM-security

Directions

- maybe tailor the security defs. $+$ model to the application, but do it sensibly (see e.g. [Boureanu Gerault Lafourcade Onete WiSec2017] for examples to the contrary)
- mechanise crypto-proofs in ... Easycrypt ?

Conclusions

THANK YOU!