Computational Indistinguishability Logics

Bruce M. Kapron
Computer Science Department
University of Victoria
bmkapron@uvic.ca

June 27, 2009

Joint work with

(CIL)

Gilles Barthe – IMDEA, Madrid

Marion Daubignard, Yassine Lahknech – VERIMAG, Universit’e de Grenoble

(Earlier work)
Russell Impagliazzo – UCSD and IAS

Motivation

✔ Well-known approaches to verification

1. Complexity-based (“provable security”)
2. Logic-based

✔ *Computationally sound logics*: Develop (2) while being faithful to (1)

✔ End of story?

- ✘ Expressivity – what security notions can we express in our logic?
- ✘ Abstraction – what is there explicitly (e.g. adversaries)?
- ✘ Model – axiomatic assumptions (e.g. OWFs), general framework (e.g. ROM)

Expressivity vs Abstraction

- ✔ We *could* formalize all of provable security in Peano arithmetic
- ✔ This seems to be worse than no formalization at all
- ✔ What makes provable security difficult?
  - ✘ Probabilities
  - ✘ Reduction paradigm
  - ✘ Modeling general computation (i.e. resource-bounded TM's)
- ✔ We want to abstract away from these details to make proofs more transparent
- ✔ We pursue an *implicit* rather than *explicit* approach

Explicit vs. Implicit

	Explicit	Implicit
Probability	$\Pr[\varphi] = p$	$\mathrm{Neg}(\varphi)$
Reduction	$\mathrm{Adv}^{S}_{\vec{p}}(\mathcal{A}) \leq \epsilon \Rightarrow \mathrm{Adv}^{P}_{\vec{q}}(R^{\mathcal{A}}) \leq \epsilon'$	$\frac{\mathrm{Sec}(P)}{\mathrm{Sec}(S)}$
Complexity	M computes f and $T_M(n) = O(n^k)$ and M uses $O(n^l)$ random bits	$f \in$ PPTF
Primitives	$\forall A \forall k \exists n_0 \forall n \geq n_0$ $\lvert \Pr[A(X_n) = 1] - \Pr[A(Y_n) = 1] \rvert$ $\leq n^{-k}$	$X \approx Y$

Explicit vs Implicit

Benefits of implicit approach:

1. Abstraction – “the devil is in the details”
2. Modularity – implicit definitions tend to be more amenable to composition, leading to more modular proofs
3. Scheme driven proofs – (“reductionist” viewpoint still possible – proof search)

But what about

Explicit vs Implicit

Benefits of implicit approach:

1. Abstraction – "the devil is in the details"
2. Modularity – implicit definitions tend to be more amenable to composition, leading to more modular proofs
3. Scheme driven proofs – ("reductionist" viewpoint still possible – proof search)

But what about

1. Concrete security
2. Parameters
3. Tightness of reductions

Extraction of concrete information from proofs is a possibility (see, e.g., the work of Kohlenbach,) as are proof "refinement" techniques.

Can't have everything!

Axiomatising ≈

- ✔ First version given by [Impagliazzo, K FOCS 2003]:
- ✔ Distribution ensembles are represented by terms for polytime functions, plus randomization operators $(\nu_{p(\eta)})$
- ✔ Axiom schema stating that $\approx$ is an equivalence relation
- ✔ (UNIV) scheme states that terms which are equal for all elements of some domain are indistinguishable define indistinguishable distributions when evaluated using randomly chosen elements of the domain
- ✔ (EDIT) scheme states that operations preserving distributional identity also preserve indisitinguishability

Axiomatising $\approx$

✔ (SUB)

$$\frac{f \approx g}{u[f/x] \approx u[g/x]}$$

(NOTE: must maintain conventions on variable capture, substitution somewhat nonstandard)

✔ (H-IND)

$$\frac{\nu_{p(\eta)} i.u[i/x] \approx \nu_{p(\eta)} i.[i+1/x]}{u[0/x] \approx u[p(\eta)/x]}$$

Hybrid induction schema; may also be formulated in a universal (as opposed to randomized) version

Axiomatising indistinguishability

- ✔ Adequate for formulating many primitive notions, e.g., f is a PRG:
$$\nu_\eta x.f(x) \approx \nu_{\eta+1} y.y$$
- ✔ Can be lifted to equivalence between functions, to formulate, e.g., PRFs ([Impagliazzo, K 2009])
- ✔ Computationally sound, but misses some obvious equivalences, e.g.,
$$\nu x.\nu r.(x, r) \approx \nu x.\nu r.(x, x \oplus r)$$
- ✔ Clearly (UNIV) may be strengthened, that in the premise universal equality may be replaced by distributional identity or statistical indistinguishability
- ✔ In this case (EDIT) is redundant

Negligible events

- ✔ $\approx$ alone is not enough to express important notions, e.g., OWF
- ✔ Need to express negligibility of events (e.g. inverting a OWF f)
- ✔ Starting point: [Halpern 08] – $\varphi \to \psi$ for $\Pr[\varphi|\psi]$ approaches 1 superpolynomially in η.
- ✔ No notion of adversary, computational element is missing – still not possible to, e.g., simply formalize OWFs

Computational Frames

- ✔ The logic of [IK03] only allows us to reason about simple distributions generated by PTT functions
- ✔ Security definitions (e.g. game-based) rely on interaction (e.g. between adversary and oracle)
- ✔ How to model this in logic?
- ✔ One approach – computational frames [AF01]

Interlude

$<$ `interlude` $>$

What we have seen so far:

1. Indistinguishability logic of [IK03]
2. Negligible (overwhelming) conditional reasoning [H08]
3. Computational Frames [AF01]

[Barthe,Daubignard,K,Lakhnech 08] uses these ideas (and several others!) to arrive at a logic capable of reasoning about a wide assortment of cryptography-base security schemes

$<$ /`interlude` $>$

Computational Frames

Consider IND-CPA game for an encryption scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$

1. Keys are generated: $(sk, pk) \overset{r}{\leftarrow} \mathcal{K}(\eta)$
2. Adversary $\mathcal{A}_1(pk)$ produces (s, m_0, m_1)
3. $b \overset{r}{\leftarrow} \{0, 1\}$
4. Adversary $\mathcal{A}_2(s, pk, m_0, m_1, \mathcal{E}(pk, m_b))$ returns b'
5. Adversary wins if $b = b'$

Computational Frames

What is the general pattern?

1. Generation of fresh random values pk, sk, b (NOTE: not all of these are "public")
2. A sequence of adversary calls are made – $\mathcal{A}_2$ can use $\mathcal{A}_1$'s output
3. Output of both adversaries depends on interaction with an oracle.

Computational Frames

This leads to a general syntax:

$$s = \vec{x}_{pu}, \vec{x}_{pr} \overset{r}{\leftarrow} G(\eta).\vec{a} \overset{r}{\leftarrow} \vec{\mathcal{A}}.(u_1, \ldots, u_m)|I_1/\mathcal{O}_1, \ldots, I_n/\mathcal{O}_n$$

Let s be a frame with p adversary variables. Fix η. Then for any sequence $\vec{\mathcal{A}} = \mathcal{A}_1, \ldots, \mathcal{A}_p, \mathcal{A}$ of poly-time adversaries, we have a distribution

$$\vec{\mathcal{A}}||s$$

Includes all (publicly) drawn x's, all a's, the resulting values of the u's, the query traces of each oracle (denoted $T\mathcal{O}_i$,) and the value returned by $\mathcal{A}$ (denoted R)

Won't define this formally in this talk

CIL Formulae – Frame indistinguishability

First type of formula

$$E \rightarrow s \sim t$$

E is an “event” (formalized in an appropriate language), s, t frames

Intendend interpretation: for any $\vec{\mathcal{A}}$

$$|\Pr[\vec{\mathcal{A}}||s.R = 1] - \Pr[\vec{\mathcal{A}}||t.R = 1]$$

is negligible in η

CIL Formulae – Conditional negligibility

For event formulas E_1, E_2 and frame s, we have a formula

$$E_2 \longrightarrow s : E_1$$

Intendended interpretation, for any $\vec{\mathcal{A}}$

$$\Pr[E_1(\vec{\mathcal{A}}||s)|E_2(\vec{\mathcal{A}}||s)]$$

is negligible in η

To say f is a OWF:

$$\nu x.f(x) : f(R) = f(x)$$

CIL Rules – Substitution

Rules for both type of formulas

$$\frac{A \rightarrow s \sim t}{A \rightarrow v[s/y] \sim v[t/y]}\text{SUB}$$

$$\frac{A \rightarrow s : E}{A \rightarrow v[s/y] : E}\text{NegSUB}$$

Clearly sound (polytime v can be composed with any adversary)

CIL Rules – Case Analysis

$$\frac{E \rightarrow s \sim t \qquad s : \neg E \qquad t : \neg E}{s \sim t}\,\mathsf{CS}$$

Idea: E holds with overwhelming probability in either frame, so conditioning on E tells us nothing

CIL Rules – External Reasoning

Introduction
Manifesto
Basic ingredients
On to CIL
Frames
CIL Formulae
CIL Rules
Applications, etc.

We have rules UNIV and NegUNIV to import reasoning about distributional equivalence (or statistical indistinguishability) into the computational setting.

Why “external” reasoning? Reasoning about e.g., distributional equivalence is essentially different – not reduction based

UCR rule relates propositional logic and conditional probability

CIL Rules – Oracles

(NOTE: we are not presenting these rules in full generality)

Let $\varphi = e \notin T\mathcal{O}_1 \wedge E$ (i.e., $\mathcal{O}_1$ is not queried at e)

$$\frac{A \rightarrow (s|I_1/\mathcal{O}_1) : \varphi \qquad q \neq e \Rightarrow I_1(q) = I'_1(q)}{A \rightarrow (s|I'_1/\mathcal{O}_1) : \varphi}\text{NegOR}\forall$$

Let $\psi = e \in T\mathcal{O}_1 \wedge E$, where E is $T\mathcal{O}$-prefix closed

$$\frac{A \rightarrow (s|I_1/\mathcal{O}_1) : \psi \qquad q \neq e \Rightarrow I_1(q) = I'_1(q)}{A \rightarrow (s|I'_1/\mathcal{O}_1) : \psi}\text{NegOR}\exists$$

$$\frac{A \rightarrow s|I_1/\mathcal{O}_1 : e \in T\mathcal{O}_1 \quad q \neq e \Rightarrow I_1(q) = I'_1(q)}{A \rightarrow s|I_1/\mathcal{O}_1 \sim s|I'/\mathcal{O}_1}\text{OR}$$

Applications, etc.

A number of security proofs (in standard model and ROM) have been formalized in CIL, including ElGamal, Hashed ElGamal, OAEP, FDH and PSS signature schemes.

Work progressing on the formalization of CIL in Coq as part of SCALP project

Future work: more support for external reasoning, extension to protocols