Computer-aided cryptographic proofs

Gilles Barthe
MPI-SP, Germany
IMDEA Software Institute, Spain

Computer-aided cryptography

Develop tool-assisted methodologies for helping the design, analysis, and implementation of cryptographic constructions (primitives and protocols)

Goals:

- Automated analysis of (symbolic or computational) security
- Independently verifiable proofs of (computational) security
- Verified implementations
- New designs and better implementations
- etc

Building on formal methods

- program analysis and verification/program synthesis
- compilation (certifying compilation/verified compilation)
- logic
- etc

Potential benefits

Formal methods for cryptography

- higher assurance
- smaller gap between provable security and crypto engineering
- new proof techniques

Cryptography for formal methods

- Challenging and non-standard examples
- New theories and applications

Challenges

- requirements: probabilistic guarantees, adversaries
- analysis: composition of two secure systems need not be secure, lack of proof methods for individual components, proofs are overly complex when methods exist
- implementation and deployment: security not preserved by refinement, legacy, standardization, side-channels

Modern cryptography

Shannon ’49

- Mathematical proof of security
- Perfect secrecy is impossible

Diffie & Hellman ’76

Goldwasser & Micali’82

Yao’82

- Computational security
- Asymptotic guarantees

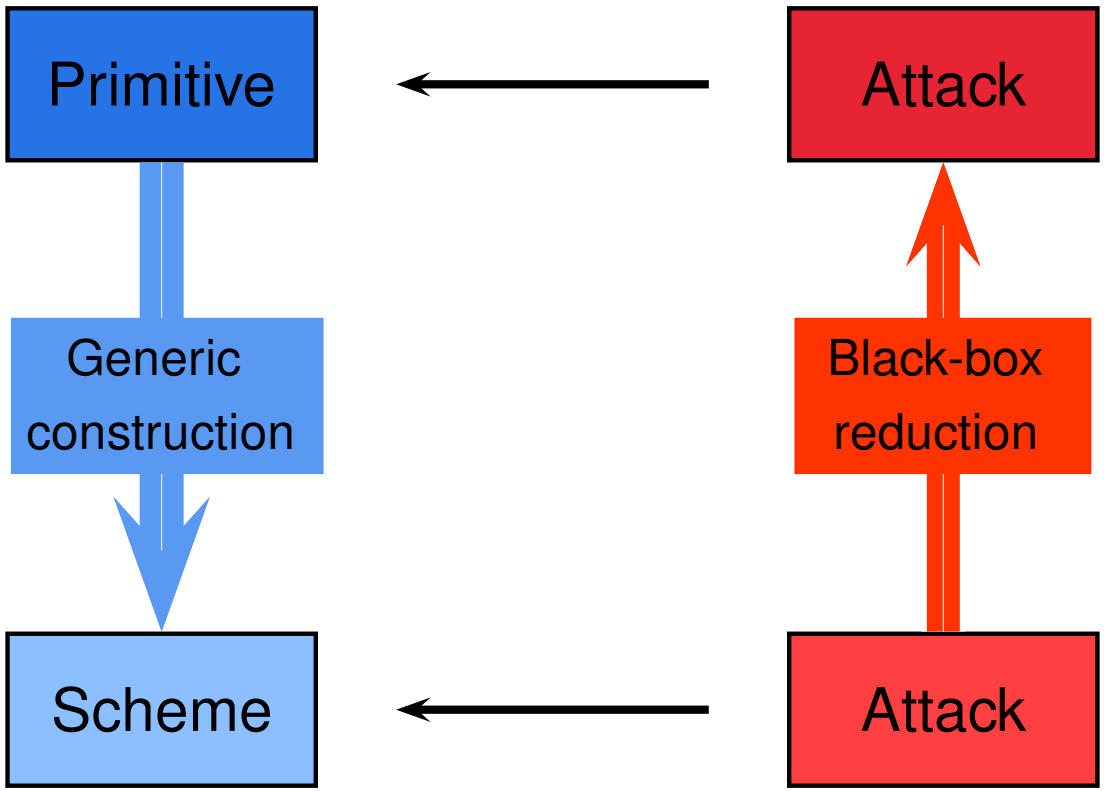
PPT adversary has negligible advantage

Bellare & Rogaway ’94

- Concrete bounds

Adversary advantage to win in time t is $\leq p$

Reductionist proof



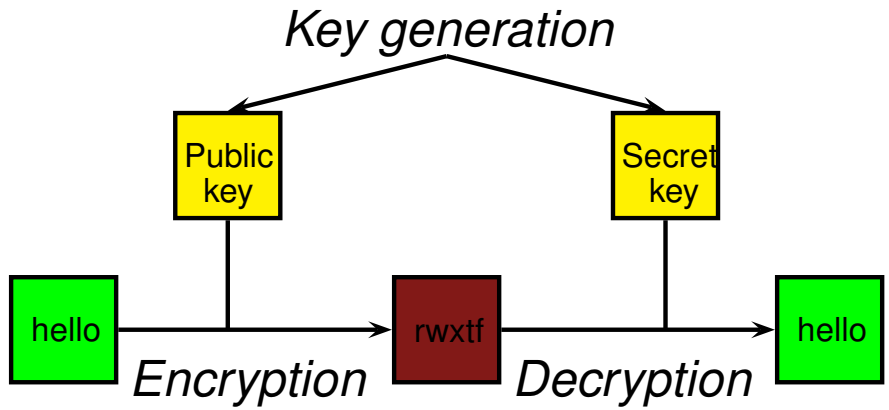
Public-key encryption

Algorithms $(\mathcal{K}, \mathcal{E}_{pk}, \mathcal{D}_{sk})$

- $\mathcal{E}$ probabilistic
- $\mathcal{D}$ deterministic and partial

If (sk, pk) is a valid key pair,

$$\mathcal{D}_{sk}(\mathcal{E}_{pk}(m)) = m$$



Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0, 1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$

Indistinguishability

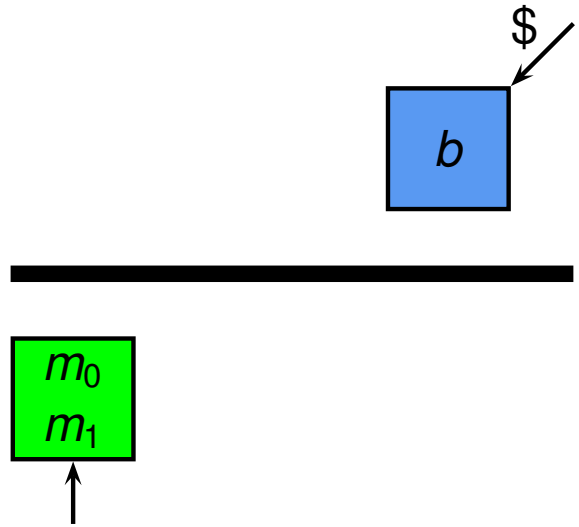
Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0, 1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$

Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0, 1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$

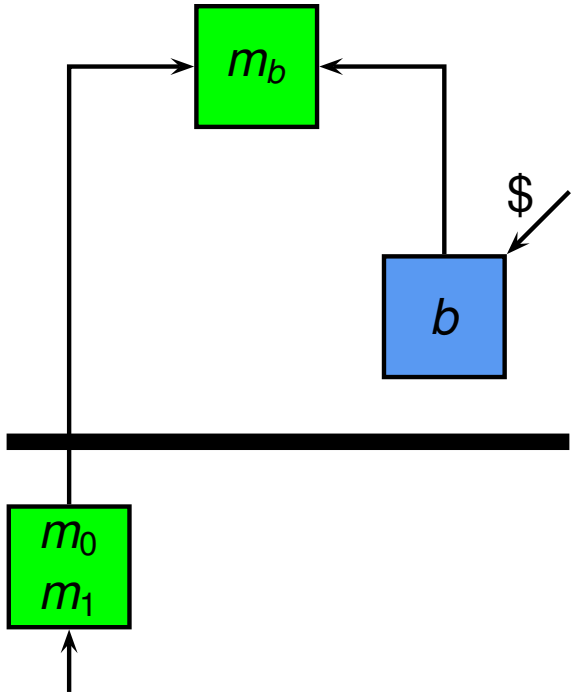
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0, 1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



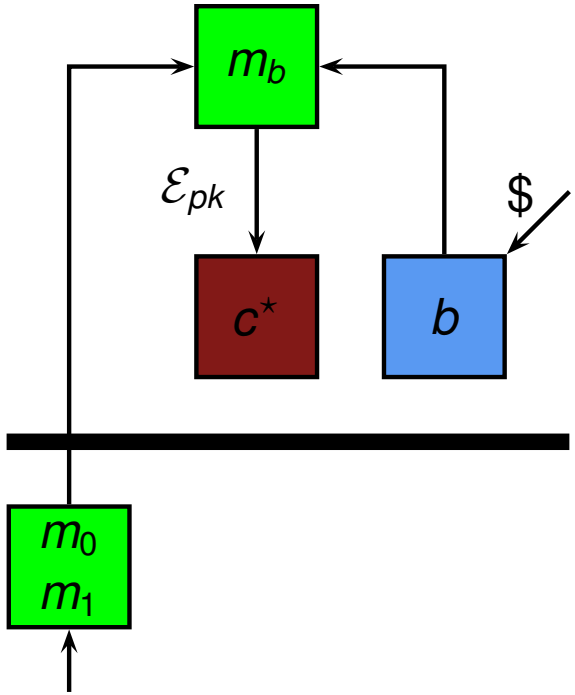
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



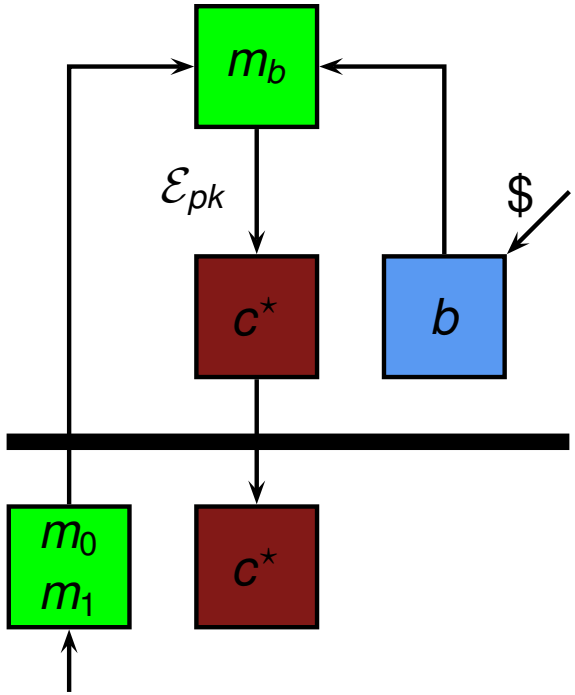
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



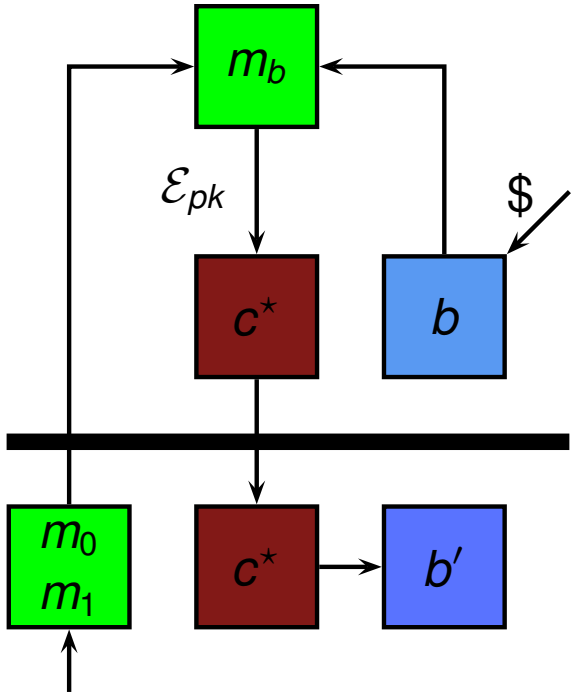
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



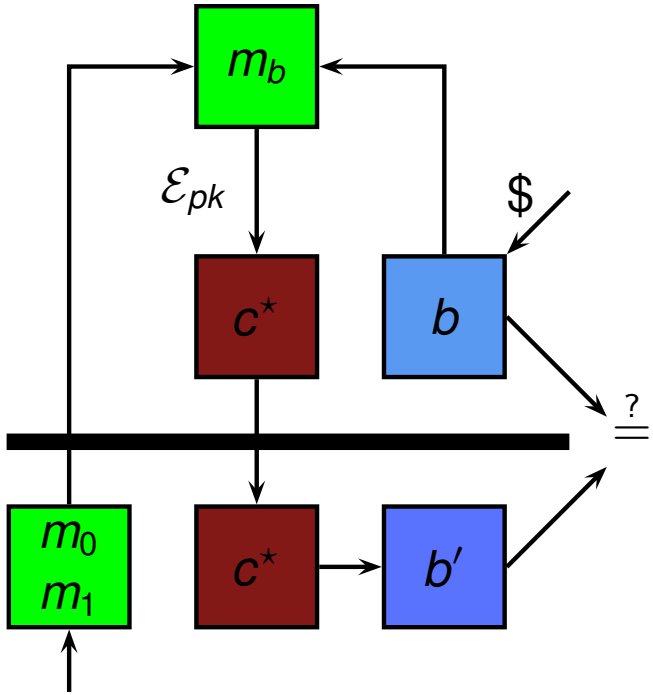
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



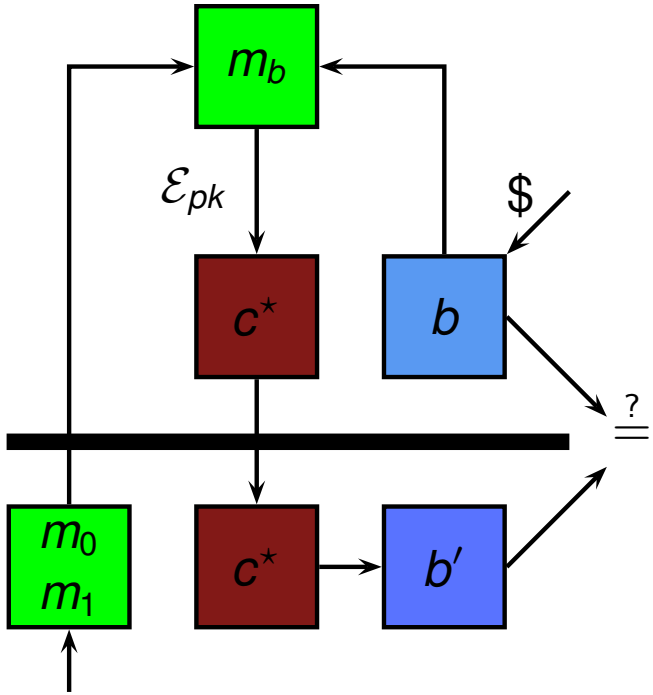
Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0, 1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



Indistinguishability

Game INDCPA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$



$$\left| \Pr_{\mathrm{INDCPA}(\mathcal{A})}\left[b' = b\right] - \frac{1}{2} \right| \quad \text{small}$$

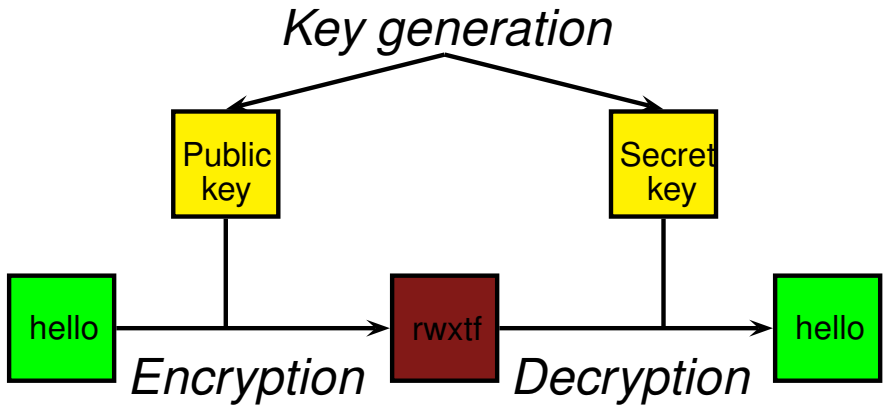
One-way trapdoor permutations

Algorithms $(\mathcal{K}, \mathsf{f}_{pk}, \mathsf{f}_{sk}^{-1})$

- f_{pk} and f_{sk}^{-1} deterministic

If (sk, pk) is a valid key pair,

$$\mathsf{f}_{sk}^{-1}(\mathsf{f}_{pk}(m)) = m$$

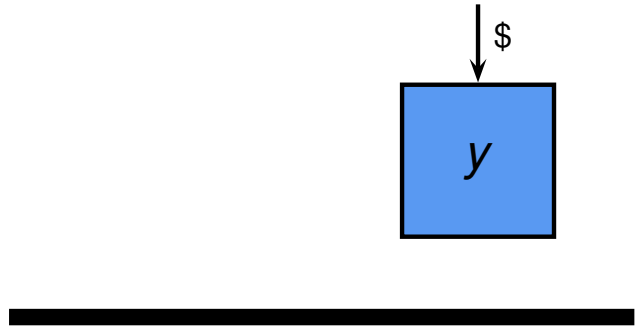


One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$

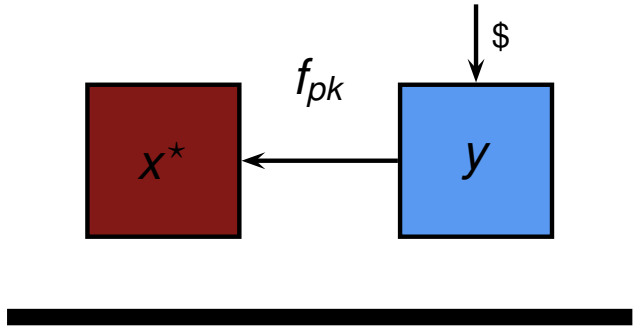
One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



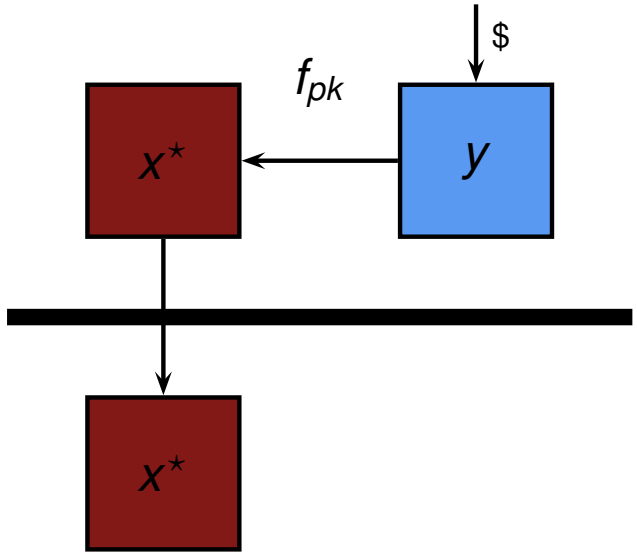
One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



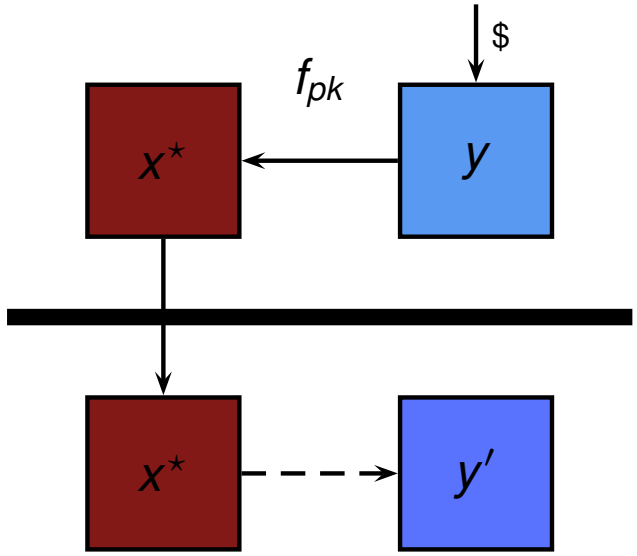
One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



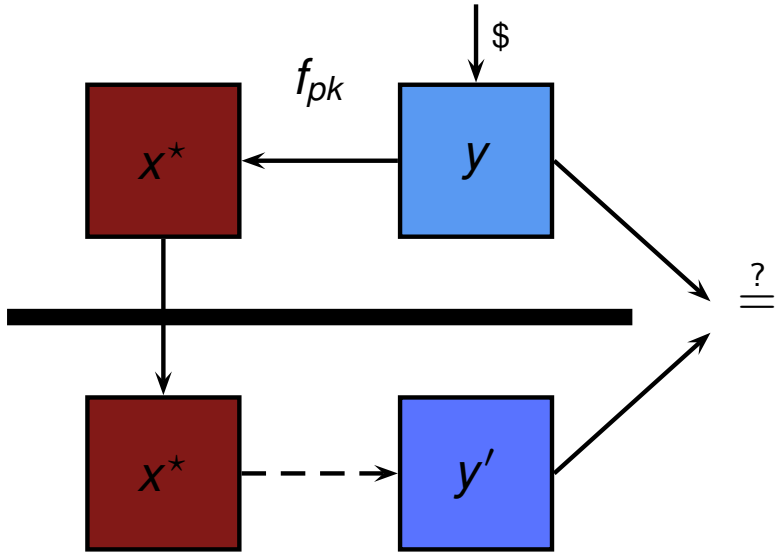
One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



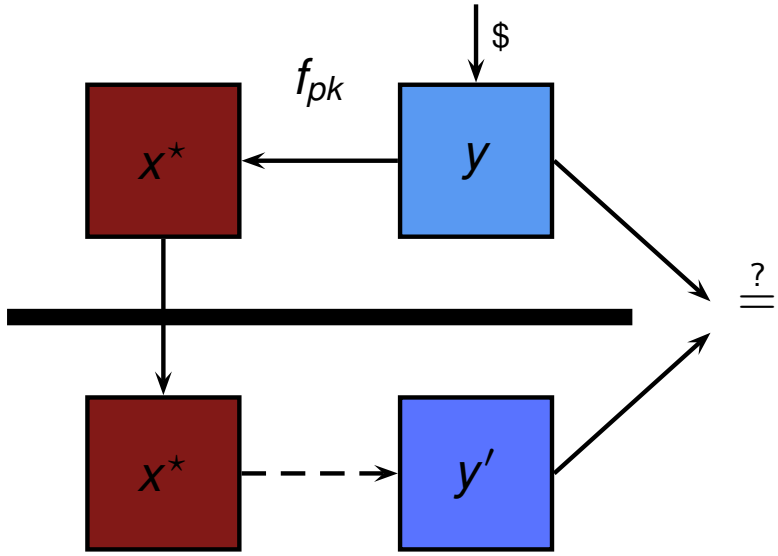
One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



One-way trapdoor permutations

Game $\mathrm{OW}(\mathcal{I})$
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^n;$
$x^\star \leftarrow \mathsf{f}_{pk}(y);$
$y' \leftarrow \mathcal{I}(x^\star);$
return $(y' = y)$



$$\Pr_{\mathrm{OW}(\mathcal{I})}\left[y' = y\right] \quad \text{small}$$

Optimal Asymmetric Encryption Padding

Encryption $\mathcal{E}_{\mathrm{OAEP}(pk)}(m)$:
$r \xleftarrow{\$} \{0,1\}^{k_0}$;
$s \leftarrow G(r) \oplus (m \| 0^{k_1})$;
$t \leftarrow H(s) \oplus r$;
return $\mathrm{f}_{pk}(s \| t)$

Decryption $\mathcal{D}_{\mathrm{OAEP}(sk)}(c)$:
$(s,t) \leftarrow \mathrm{f}_{sk}^{-1}(c)$;
$r \leftarrow t \oplus H(s)$;
if $([s \oplus G(r)]^{k_1} = 0^{k_1})$
then $m \leftarrow [s \oplus G(r)]_k$
else$m \leftarrow \perp$;
return m

Oracle $H(x)$:
if $x \notin L$ then
$r \xleftarrow{\$} \{0,1\}^{k}$;
$L \leftarrow (x,r) :: L$;
return $L[x]$;

$\oplus$ exclusive or $\|$concatenation $[\cdot]$ projection 0 zero bitstring

OAEP: provable security

Game INDCCA($\mathcal{A}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$

Game SPDOW($\mathcal{I}$)
$(sk, pk) \leftarrow \mathcal{K}()$;
$y \xleftarrow{\$} \{0,1\}^{k_2}$; $z \xleftarrow{\$} \{0,1\}^{k_3}$;
$x^\star \leftarrow \mathsf{f}_{pk}(y \,\|\, z)$;
$Y' \leftarrow \mathcal{I}(x^\star)$;
return $(y \in Y')$

FOR ALL IND-CCA adversary $\mathcal{A}$ against $(\mathcal{K}, \mathcal{E}_{\mathrm{OAEP}}, \mathcal{D}_{\mathrm{OAEP}})$,
THERE EXISTS a SPDOW adversary $\mathcal{I}$ against $(\mathcal{K}, \mathsf{f}, \mathsf{f}^{-1})$ st

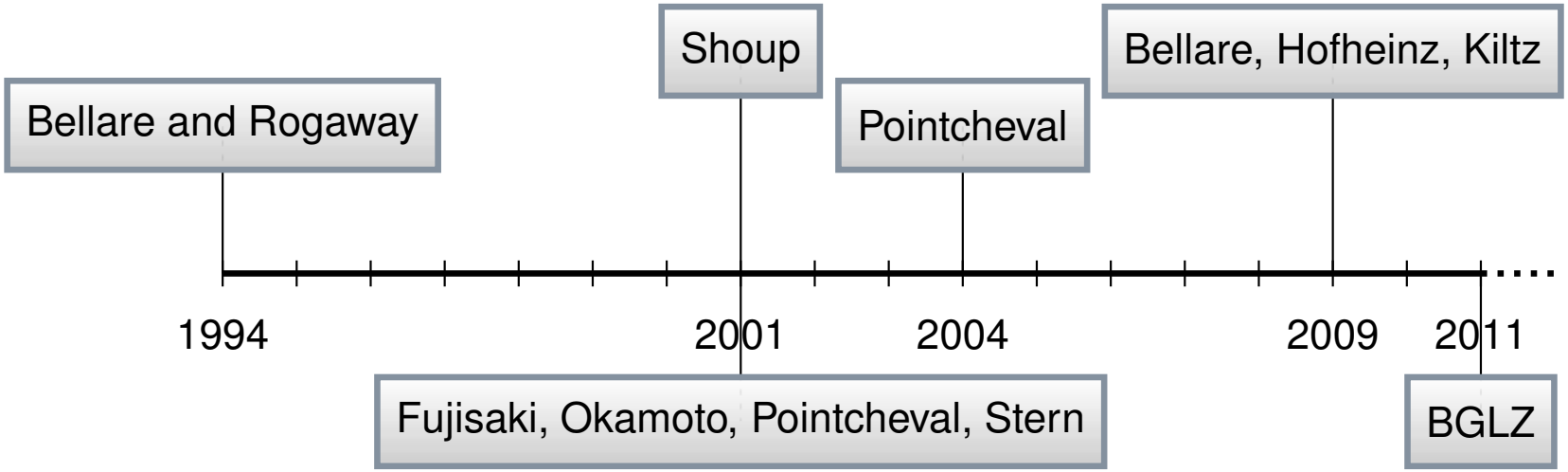
$$\left|\mathrm{Pr}_{\text{IND-CCA}(\mathcal{A})}[b' = b] - \tfrac{1}{2}\right| \leq$$

$$\mathrm{Pr}_{\mathrm{SPDOW}(\mathcal{I})}[y \in Y'] + \frac{3q_D q_G + q_D^2 + 4q_D + q_G}{2^{k_0}} + \frac{2q_D}{2^{k_1}}$$

and

$$t_{\mathcal{I}} \leq t_{\mathcal{A}} + q_D\, q_G\, q_H\, T_{\mathsf{f}}$$

OAEP: provable security



1994 Purported proof of chosen-ciphertext security

2001 1994 proof gives weaker security; desired security holds

- for a modified scheme
- under stronger assumptions

2004 Filled gaps in 2001 proof

2009 Security definition needs to be clarified

2011 Fills gaps in 2004 proof

Example: Bellare and Rogaway 1993 encryption

Game INDCPA($\mathcal{A}$) :
$(sk, pk) \leftarrow \mathcal{K}(\ )$;
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk)$;
$b \xleftarrow{\$} \{0,1\}$;
$c^\star \leftarrow \mathcal{E}_{pk}(m_b)$;
$b' \leftarrow \mathcal{A}_2(c^\star)$;
return $(b' = b)$

Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell$;
$s \leftarrow H(r) \oplus m$;
$y \leftarrow f_{pk}(r) \,\|\, s$;
return y

For every adversary $\mathcal{A}$, there exists an inverter $\mathcal{I}$ st

$$\left| \Pr_{\mathrm{INDCPA}(\mathcal{A})}\left[b' = b\right] - \frac{1}{2} \right| \leq \Pr_{\mathrm{OW}(\mathcal{I})}\left[y' = y\right]$$

Proof

Game hopping technique

Game INDCPA :
$(sk, pk) \leftarrow \mathcal{K}();$
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$c^\star \leftarrow \mathcal{E}_{pk}(m_b);$
$b' \leftarrow \mathcal{A}_2(c^\star);$
return $(b' = b)$
Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell;$
$h \leftarrow H(r);$
$s \leftarrow h \oplus m;$
$c \leftarrow f_{pk}(r) \,\|\, s;$
return c

Game G :
$(sk, pk) \leftarrow \mathcal{K}();$
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$c^\star \leftarrow \mathcal{E}_{pk}(m_b);$
$b' \leftarrow \mathcal{A}_2(c^\star);$
return $(b' = b)$
Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell;$
$h \xleftarrow{\$} \{0,1\}^k;$
$s \leftarrow h \oplus m;$
$c \leftarrow f_{pk}(r) \,\|\, s;$
return c

Game G′ :
$(sk, pk) \leftarrow \mathcal{K}();$
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$c^\star \leftarrow \mathcal{E}_{pk}(m_b);$
$b' \leftarrow \mathcal{A}_2(c^\star);$
return $(b' = b)$
Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell;$
$s \xleftarrow{\$} \{0,1\}^k;$
$h \leftarrow s \oplus m;$
$c \leftarrow f_{pk}(r) \,\|\, s;$
return c

Game OW :
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^\ell;$
$y' \leftarrow \mathcal{I}(f_{pk}(y));$
return $y = y'$
Adversary $\mathcal{I}(x)$:
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$s \xleftarrow{\$} \{0,1\}^k;$
$c^\star \leftarrow x \,\|\, s;$
$b' \leftarrow \mathcal{A}_2(c^\star);$
$y' \leftarrow [z \in L_{\mathcal{A}} \,|\, f_{pk}(z) = x];$
returny'

1. For each hop
   - prove validity of pRHL judgment
   - derive probability claims
   - (possibly) resolve some probability expressions using pHL
2. Obtain security bound by combining claims
3. Check execution time of constructed adversary

Conditional equivalence

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$h \leftarrow H(r)$;
$s \leftarrow h \oplus m$;
$c \leftarrow f_{pk}(r) \,\|\, s$;
return c

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$h \xleftarrow{\$} \{0,1\}^{k}$;
$s \leftarrow h \oplus m$;
$c \leftarrow f_{pk}(r) \,\|\, s$;
return c

$$\vDash \{\top\}\ \mathsf{INDCPA}\ \sim\ \mathbf{G}\ \left\{(\neg r \in L_{\mathcal{A}})\langle 2\rangle\ \rightarrow =_{b,b'}\right\}$$

$$\left|\Pr_{\mathsf{INDCPA}}\left[b' = b\right] - \Pr_{\mathbf{G}}\left[b' = b\right]\right| \leq \Pr_{\mathbf{G}}[r \in L_{\mathcal{A}}]$$

Equivalence

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$h \xleftarrow{\$} \{0,1\}^{k}$;
$s \leftarrow h \oplus m$;
$c \leftarrow f_{pk}(r) \| s$;
return c

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$s \xleftarrow{\$} \{0,1\}^{k}$;
$h \leftarrow s \oplus m$;
$c \leftarrow f_{pk}(r) \| s$;
return c

$$\models \{\top\}\ \mathbf{G}\ \sim\ \mathbf{G}'\ \{=_{b,b',r,\mathcal{A}}\}$$

$$\Pr_{\mathbf{G}}[r \in L_{\mathcal{A}}] = \Pr_{\mathbf{G}'}[r \in L_{\mathcal{A}}] \qquad \Pr_{\mathbf{G}}[b' = b] = \Pr_{\mathbf{G}'}[b' = b] = \frac{1}{2}$$

Equivalence

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$h \xleftarrow{\$} \{0,1\}^{k}$;
$s \leftarrow h \oplus m$;
$c \leftarrow f_{pk}(r) \| s$;
return c

$\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^{\ell}$;
$s \xleftarrow{\$} \{0,1\}^{k}$;
$h \leftarrow s \oplus m$;
$c \leftarrow f_{pk}(r) \| s$;
return c

$$\vDash \{\top\}\ \mathbf{G}\ \sim\ \mathbf{G}'\ \{=_{b,b',r,\mathcal{A}}\}$$

$$\left|\Pr_{\mathsf{INDCPA}}[b' = b] - \tfrac{1}{2}\right| \leq \Pr_{\mathbf{G}'}[r \in L_{\mathcal{A}}]$$

Reduction

Game INDCPA :
$(sk, pk) \leftarrow \mathcal{K}();$
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$c^\star \leftarrow \mathcal{E}_{pk}(m_b);$
$b' \leftarrow \mathcal{A}_2(c^\star);$
return $(b' = b)$
Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell;$
$s \xleftarrow{\$} \{0,1\}^k;$
$c \leftarrow f_{pk}(r) \,\|\, s;$
return c

Game OW :
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^\ell;$
$y' \leftarrow \mathcal{I}(f_{pk}(y));$
return $y = y'$
Adversary $\mathcal{I}(x)$:
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$s \xleftarrow{\$} \{0,1\}^k;$
$c^\star \leftarrow x \,\|\, s;$
$b' \leftarrow \mathcal{A}_2(c^\star);$
$y' \leftarrow [z \in L_{\mathcal{A}} \mid f_{pk}(z) = x];$
returny'

$$\vDash \{\top\}\ \mathbf{G}'\ \sim\ \mathrm{OW}\ \left\{(r \in L_{\mathcal{A}})\langle 1\rangle \rightarrow (y' = y)\langle 2\rangle\right\}$$

$$\Pr_{\mathbf{G}'}[r \in L_{\mathcal{A}}] \leq \Pr_{\mathrm{OW}(\mathcal{I})}[y' = y]$$

Reduction

Game INDCPA :
$(sk, pk) \leftarrow \mathcal{K}();$
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$c^\star \leftarrow \mathcal{E}_{pk}(m_b);$
$b' \leftarrow \mathcal{A}_2(c^\star);$
return $(b' = b)$
Encryption $\mathcal{E}_{pk}(m)$:
$r \xleftarrow{\$} \{0,1\}^\ell;$
$s \xleftarrow{\$} \{0,1\}^k;$
$c \leftarrow f_{pk}(r) \,\|\, s;$
return c

Game OW :
$(sk, pk) \leftarrow \mathcal{K}();$
$y \xleftarrow{\$} \{0,1\}^\ell;$
$y' \leftarrow \mathcal{I}(f_{pk}(y));$
return $y = y'$
Adversary $\mathcal{I}(x)$:
$(m_0, m_1) \leftarrow \mathcal{A}_1(pk);$
$b \xleftarrow{\$} \{0,1\};$
$s \xleftarrow{\$} \{0,1\}^k;$
$c^\star \leftarrow x \,\|\, s;$
$b' \leftarrow \mathcal{A}_2(c^\star);$
$y' \leftarrow [z \in L_{\mathcal{A}} \mid f_{pk}(z) = x];$
returny'

$$\vDash \{\top\}\ \mathbf{G}'\ \sim\ \mathrm{OW}\ \{(r \in L_{\mathcal{A}})\langle 1\rangle \rightarrow (y' = y)\langle 2\rangle\}$$

$$\left|\mathrm{Pr}_{\mathrm{INDCPA}\ (\mathcal{A})}[b' = b] - \tfrac{1}{2}\right| \leq \mathrm{Pr}_{\mathrm{OW}(\mathcal{I})}[y' = y]$$

Automated proofs

$$f((m\|0) \oplus G(r) \ \| r \oplus H((m\|0) \oplus G(r)))$$

- Hard to get security proofs right
- 6 months to formalize the proof!
- Many variants in the literature
- About 200 variants of SAEP/OAEP (Komano and Ohta)
- About $10^6 - 10^8$ candidates schemes of “reasonable” size
- Can we automate analysis for finding attacks or proofs?

ZooCrypt

- Extremely efficient logics for CPA and CCA security (up-to-bad, optimistic sampling, reduction, reject some ciphertexts)
- Extremely efficient procedures for detecting attacks
- Smart generation of candidate constructions

Experiments

- Generated 1,000,000 candidates
- For CPA security: 99,5% solved by the tool
- For CCA security: 80% solved by tool
- Practical interpretation (sql database)
- Manual inspection for grey zone
- Interactive tutor

ZAEP

- OAEP (1994):

$$f((m\|0) \oplus G(r) \ \| \ r \oplus H((m\|0) \oplus G(r)))$$

- SAEP (2001):

$$f(r \ \| \ (m\|0) \oplus G(r))$$

- ZAEP (2012):

$$f(r \ \| \ m \oplus G(r))$$

  - ☞ redundancy-free
  - ☞ INDCCA secure for RSA with exponent 2 and 3

Other automated tools

- Auto G&P: pairing-based crypto
- GGA: pairing-based crypto
- AutoLWE: lattice-based crypto
- Blockciphers

Need general purpose tools

EasyCrypt

Domain-specific proof assistant

- proof goals tailored to reductionist proofs
- proof tools support common proof techniques (bridging steps, failure events, hybrid arguments, eager sampling. . .)

Control and automation from state-of-art verification

- interactive proof engine and mathematical libraries (a la Coq/ssreflect)
- back-end to SMT solvers

Many case studies:

- Encryption, signatures, key exchange, zero-knowledge, multi-party and verifiable computation, SHA3, voting, KMS

probabilistic relational Hoare logic

- Code-based approach

$$
\begin{array}{lcll}
\mathcal{C} & ::= & \mathsf{Skip} & \text{skip} \\
& | & \mathcal{V} \leftarrow \mathcal{E} & \text{assignment} \\
& | & \mathcal{V} \xleftarrow{\$} \mathcal{D} & \text{random sampling} \\
& | & \mathcal{C}; \mathcal{C} & \text{sequence} \\
& | & \text{if } \mathcal{E} \text{ then } \mathcal{C} \text{ else } \mathcal{C} & \text{conditional} \\
& | & \text{while } \mathcal{E} \text{ do } \mathcal{C} & \text{while loop} \\
& | & \mathcal{V} \leftarrow \mathcal{F}(\mathcal{E}, \ldots, \mathcal{E}) & \text{procedure (oracle/adv) call}
\end{array}
$$

- Game-playing technique: $\models \{P\}\ c_1 \sim c_2\ \{Q\}$ where P and Q are relations on states

Let $\mu_1, \mu_2 \in \mathrm{Dist}(A)$ and $R \subseteq A \times A$. Let $\mu \in \mathrm{Dist}(A \times A)$.

- μ is a coupling for (μ_1, μ_2) iff $\pi_1(\mu) = \mu_1$ and $\pi_2(\mu) = \mu_2$
- μ is a R-coupling for (μ_1, μ_2) if moreover $\Pr_{y \leftarrow \mu}[y \notin R] = 0$

Verified implementations

- FOR EVERY adversary that breaks assembly code,
- IF assembly code is safe and leakage resistent,
- AND assembly code correctly implements algorithm,
- THERE EXISTS an adversary that breaks the algorithm

Summary

Foundations and tools for high-assurance cryptography

- Provable security
- Practical cryptography
- Reducing the gap between security proofs and implementations

Methods apply to

- Differential privacy
- Machine Learning