Indifferentiability and Security Proofs in Idealized Models

Yannick Seurin
Orange Labs
yannick.seurin@orange-ftgroup.com

21 May 2010
Univ. Rennes Crypto Seminar

intro

- unconditional security (a.k.a. information-theoretic security): considers computationally unbounded adversaries, very inefficient schemes
- standard model: polynomially-bounded adversaries, relies on complexity assumptions, most desirable framework
- idealised models (ROM, ICM…): good guideline to design efficient schemes
- heuristic arguments and proof against specific attacks (e.g. proof that AES is immune to differential and linear cryptanalysis)
- security proofs are never absolute: they rely on an attack model and usually computational assumptions

outline

indifferentiability

equivalence of the ROM and the ICM

doubling the domain of an ideal cipher

outline

indifferentiability

equivalence of the ROM and the ICM

doubling the domain of an ideal cipher

the random oracle model (ROM)

- modelizes a perfect hash function
- *Random Oracle Model* [BellareR93]: a publicly accessible oracle, returning a n-bit random value for each new query
- widely used in PK security proofs (OAEP, PSS...)
- uninstantiability results [CanettiGH98, Nielsen02]
- schemes provably secure in the plain standard model
  - Cramer-Shoup encryption
  - Boneh-Boyen signatures...

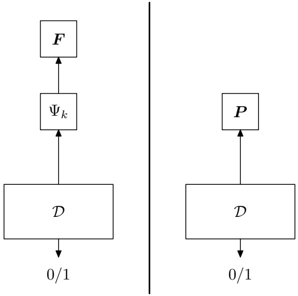
  are often less efficient or come at the price of less standard complexity assumptions

the ideal cipher model (ICM) and the random permutation model

- ICM modelizes a perfect a block cipher [Shannon49, Winternitz84]
- *Ideal Cipher Model*: a pair of publicly accessible oracles $\boldsymbol{E}(\cdot,\cdot)$ and $\boldsymbol{E}^{-1}(\cdot,\cdot)$, such that $\boldsymbol{E}(K,\cdot)$ is a random permutation for each key K
- *Random Permutation Model*: a single random permutation oracle $\boldsymbol{P}$ and its inverse $\boldsymbol{P}^{-1}$
- less popular than the ROM, but:
  - widely used for analyzing block cipher-based hash functions [BlackRS02, Hirose06]
  - used for the security proof of some PK schemes (encryption, Authenticated Key Exchange. . .)
- uninstantiability results as well [Black06]

the "classical" indistinguishability notion

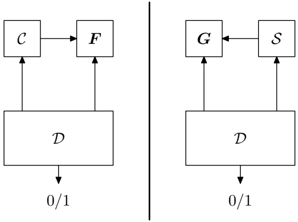
- well-known Luby-Rackoff result: the Feistel scheme with 3 (resp. 4) rounds and random functions is indistinguishable from a random permutation (resp. invertible RP)
- $\Rightarrow$ any cryptosystem proven secure with a random permutation remains secure with the LR construction and **secret** random functions



- useful only in secret-key applications (e.g. PRF to PRP conversion)
- how do we generalise indistinguishability when the internal functions are **public**? (*e.g.* for block cipher-based hash functions, public-key encryption...)

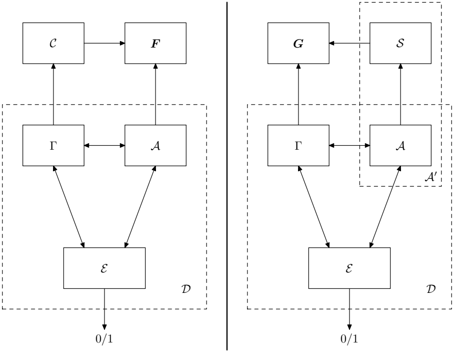
indifferentiability: definition [MRH04]

- let $\boldsymbol{G}$ be an ideal primitive (*e.g.* a random permutation), and $\mathcal{C}$ be a construction using another ideal primitive $\boldsymbol{F}$ which is **public** (*e.g.* the Feistel construction using a random oracle)



- $\mathcal{C}^{\boldsymbol{F}}$ is said to be (q, σ, ϵ)-indifferentiable from $\boldsymbol{G}$ if there is a simulator $\mathcal{S}$ making σ queries to $\boldsymbol{G}$ and such that any $\mathcal{D}$ making at most q queries distinguishes $(\mathcal{C}^{\boldsymbol{F}}, \boldsymbol{F})$ and $(\boldsymbol{G}, \mathcal{S}^{\boldsymbol{G}})$ with advantage at most ϵ
- informally the answers of $\mathcal{S}$ must be:
  - consistent with answers the distinguisher can obtain directly from $\boldsymbol{G}$
  - indistinguishable from random
- the simulator cannot see the distinguisher's queries to $\boldsymbol{G}$!

indifferentiability is the right notion



- any attacker against a cryptosystem Γ using $\mathcal{C}^{\boldsymbol{F}}$ can be turned into an attacker against Γ using $\boldsymbol{G}$ by combining the attacker with the simulator
- $\Rightarrow \mathcal{C}^{\boldsymbol{F}}$ can replace $\boldsymbol{G}$ in any cryptosystem without loss of security

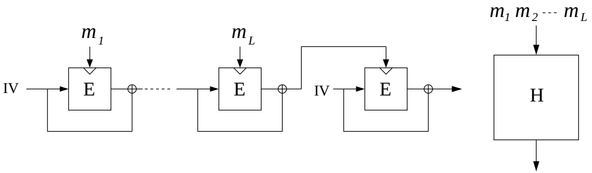
outline

indifferentiability

equivalence of the ROM and the ICM

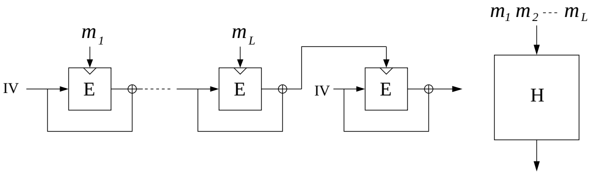
doubling the domain of an ideal cipher

the ICM implies the ROM



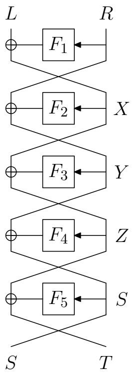
- the ideal cipher model implies the random oracle model [CDMP05]
- variants of Merkle-Damgård used with an ideal cipher in Davies-Meyer mode is indifferentiable from a random oracle
- $\Rightarrow$ the construction can replace a RO in any cryptosystem without loss of security
- what about the other direction?

the ICM implies the ROM



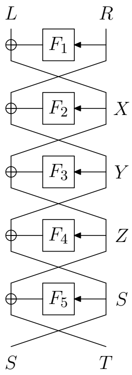
- the ideal cipher model implies the random oracle model [CDMP05]
- variants of Merkle-Damgård used with an ideal cipher in Davies-Meyer mode is indifferentiable from a random oracle
- $\Rightarrow$ the construction can replace a RO in any cryptosystem without loss of security
- what about the other direction? $\rightarrow$ Luby-Rackoff with 6 rounds

5 rounds are not enough [CoronJP02]



5 rounds are not enough [CoronJP02]

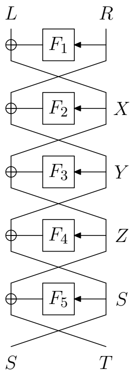
X_{12}



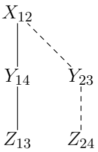
5 rounds are not enough [CoronJP02]

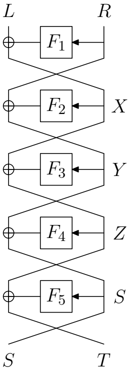
X_{12}

Y_{14} Y_{23}

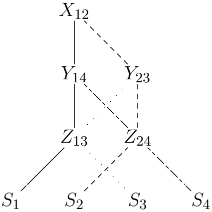


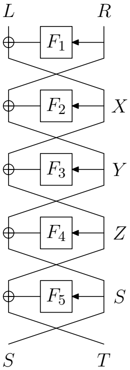
5 rounds are not enough [CoronJP02]



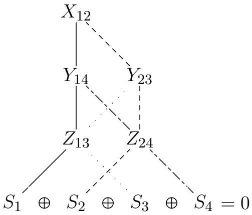


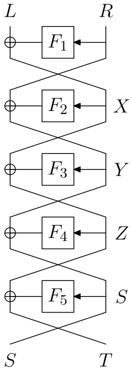
5 rounds are not enough [CoronJP02]



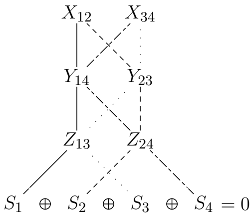


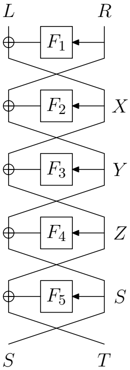
5 rounds are not enough [CoronJP02]



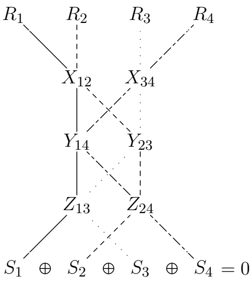


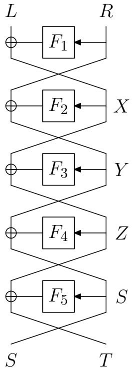
5 rounds are not enough [CoronJP02]



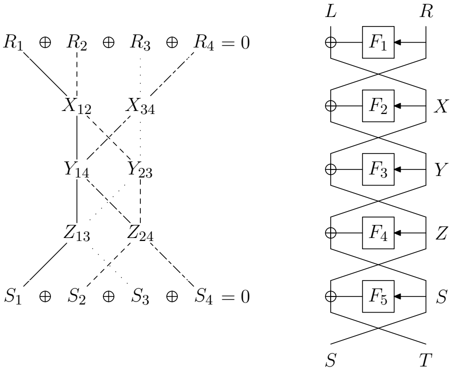


5 rounds are not enough [CoronJP02]

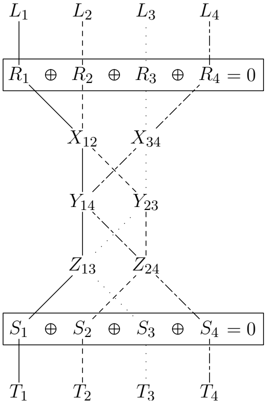


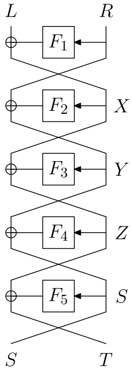


5 rounds are not enough [CoronJP02]

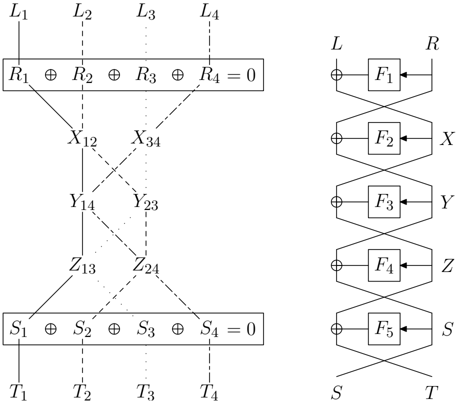


5 rounds are not enough [CoronJP02]



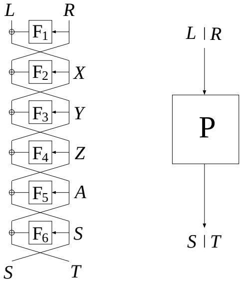


5 rounds are not enough [CoronJP02]



- for a random permutation one cannot find four I/O pairs such that $R_0 \oplus R_1 \oplus R_2 \oplus R_3 = 0$ and $S_0 \oplus S_1 \oplus S_2 \oplus S_3 = 0$ except with negl. prob.

indifferentiability for 6 rounds or more



Theorem

The Luby-Rackoff construction with 6 rounds is (q, σ, ϵ)-indifferentiable from a random permutation, with $\sigma = \mathcal{O}(q^8)$ and $\epsilon = \mathcal{O}(q^{16}/2^n)$.

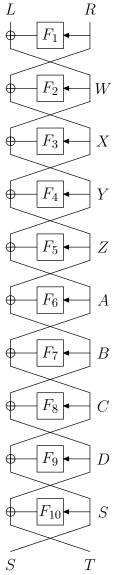
- prepending a k-bit key to the random oracle calls yields a construction indifferentiable from an ideal cipher
- simpler proof for 10 rounds (and better bounds):

Theorem

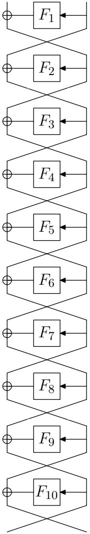
The Luby-Rackoff construction with 10 rounds is (q, σ, ϵ)-indifferentiable from a random permutation, with $\sigma = \mathcal{O}(q^4)$ and $\epsilon = \mathcal{O}(q^4/2^n)$.

simulation strategy

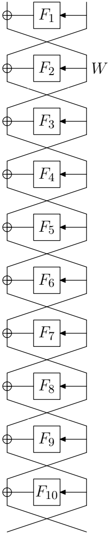
- the simulator maintains an history for each F_i with
  - values previously answered to the distinguisher
  - values defined "by anticipation"
- when a query is not in the history, $F_i(U)$ is defined randomly
- the simulator completes "chains" created in the history:
  - external chains (W, R, S, D)
  - centers (Z, A)



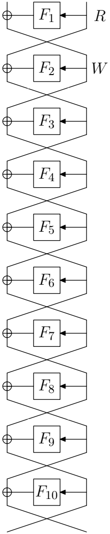
simulation strategy: external chains



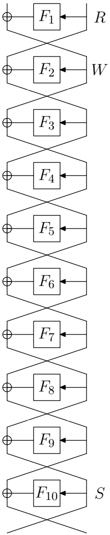
simulation strategy: external chains



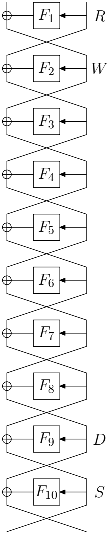
simulation strategy: external chains



simulation strategy: external chains



simulation strategy: external chains

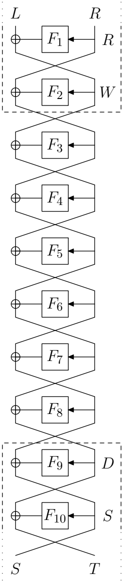


simulation strategy: external chains

- when W, R, S, D are such that

$$\boldsymbol{P}((W \oplus F_1(R)) \| R) = S \| (D \oplus F_{10}(S))$$

  they form an *external chain*

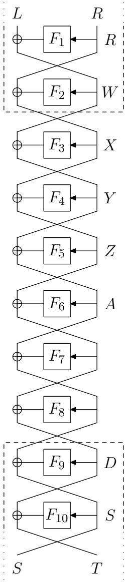


simulation strategy: external chains

- when W, R, S, D are such that

  $$\boldsymbol{P}((W \oplus F_1(R)) \| R) = S \| (D \oplus F_{10}(S))$$

  they form an *external chain*
- the simulator completes the chain, defining $F_3(X)$, $F_4(Y)$, $F_5(Z)$ and $F_6(A)$ randomly...



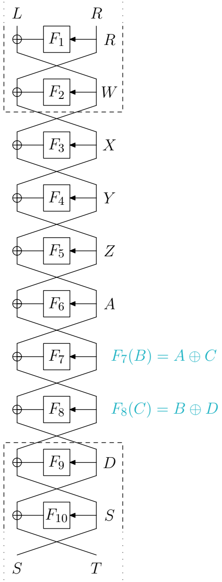
simulation strategy: external chains

- when W, R, S, D are such that

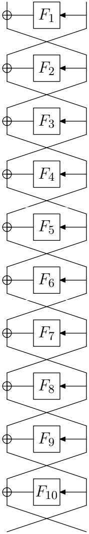
$$\boldsymbol{P}((W \oplus F_1(R))\|R) = S\|(D \oplus F_{10}(S))$$

  they form an *external chain*
- the simulator completes the chain, defining $F_3(X)$, $F_4(Y)$, $F_5(Z)$ and $F_6(A)$ randomly...
- ...and adapts the values of $F_7(B)$ and $F_8(C)$ so that

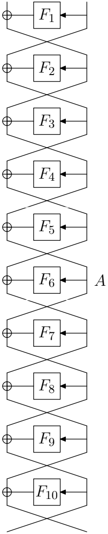
$$\Psi_{10}(L\|R) = \boldsymbol{P}(L\|R)$$



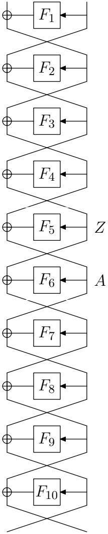
simulation strategy: centers



simulation strategy: centers

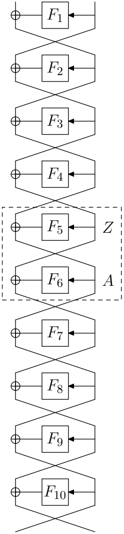


simulation strategy: centers



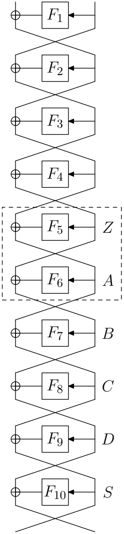
simulation strategy: centers

- any two values A and Z form a *center*



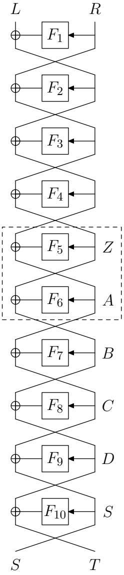
simulation strategy: centers

- any two values A and Z form a *center*
- the simulator defines $F_7(B)$, $F_8(C)$, $F_9(D)$, and $F_{10}(S)$ randomly...



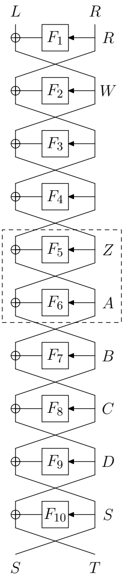
simulation strategy: centers

- any two values A and Z form a *center*
- the simulator defines $F_7(B)$, $F_8(C)$, $F_9(D)$, and $F_{10}(S)$ randomly...
- ...calls $\boldsymbol{P}^{-1}(S \| (D \oplus F_{10}(S))) = L \| R$...



simulation strategy: centers

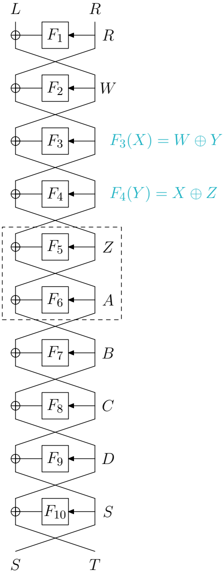
- any two values A and Z form a *center*
- the simulator defines $F_7(B)$, $F_8(C)$, $F_9(D)$, and $F_{10}(S)$ randomly...
- ...calls $\boldsymbol{P}^{-1}(S \| (D \oplus F_{10}(S))) = L \| R$...
- ...defines randomly $F_1(R)$ and $F_2(W)$...



simulation strategy: centers

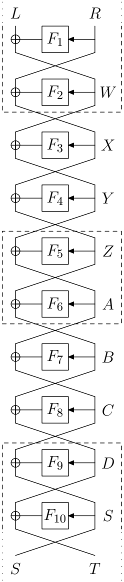
- any two values A and Z form a *center*
- the simulator defines $F_7(B)$, $F_8(C)$, $F_9(D)$, and $F_{10}(S)$ randomly...
- ...calls $\boldsymbol{P}^{-1}(S\|(D \oplus F_{10}(S))) = L\|R$...
- ...defines randomly $F_1(R)$ and $F_2(W)$...
- ...and adapts the values of $F_3(X)$ and $F_4(Y)$ so that

$$\Psi_{10}(L\|R) = \boldsymbol{P}(L\|R)$$



what could go wrong

- *exponential running-time*
  - completion of external chains creates new centers. . .
  - . . . completion of centers creates new external chains. . .
  - etc. . .
- *impossibility to adapt*
  - if the value that the simulator wants to adapt is already in the history, the simulator aborts. . .

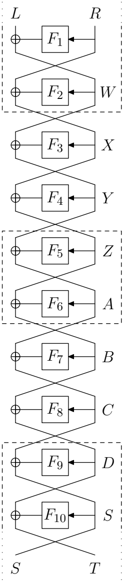


sketch of the proof

- one must show that:
  - the simulator runs in polynomial time (no "chain reaction" leading to exponentially many recursive chain completions)
  - the simulator does not have to adapt values already in the history
  - the two systems $(\Psi_{10}^{\boldsymbol{F}}, \boldsymbol{F})$ and $(\boldsymbol{P}, \mathcal{S}^{\boldsymbol{P}})$ are indistinguishable

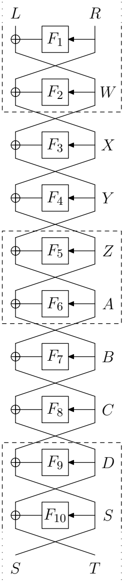
the simulator runs in polynomial time

- comes from the fact that an external chain is created with non-negligible probability only if the distinguisher has made the corresponding query $\boldsymbol{P}(L \| R) = S \| T$ or $\boldsymbol{P}^{-1}(S \| T) = L \| R$
  $\Rightarrow$ this number is less than q



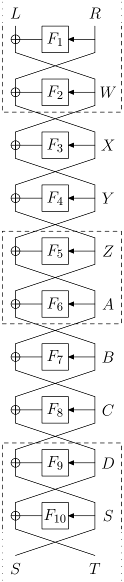
the simulator runs in polynomial time

- comes from the fact that an external chain is created with non-negligible probability only if the distinguisher has made the corresponding query $\boldsymbol{P}(L\|R) = S\|T$ or $\boldsymbol{P}^{-1}(S\|T) = L\|R$
  $\Rightarrow$ this number is less than q
- implies in turn that the history of F_5 and F_6 is bounded by $2q$
  $\Rightarrow$ the number of centers is less than $4q^2$

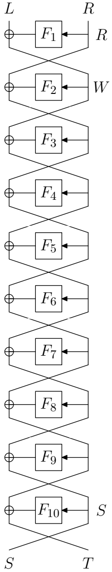


the simulator runs in polynomial time

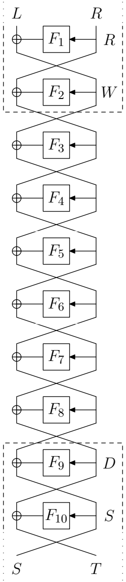
- comes from the fact that an external chain is created with non-negligible probability only if the distinguisher has made the corresponding query $\boldsymbol{P}(L\|R) = S\|T$ or $\boldsymbol{P}^{-1}(S\|T) = L\|R$
  $\Rightarrow$ this number is less than q
- implies in turn that the history of F_5 and F_6 is bounded by $2q$
  $\Rightarrow$ the number of centers is less than $4q^2$
- leads to a number of $\boldsymbol{P}$-queries of the simulator $\mathcal{O}(q^4)$



the simulator can always adapt

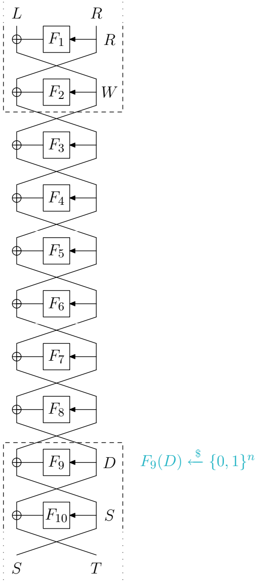


the simulator can always adapt



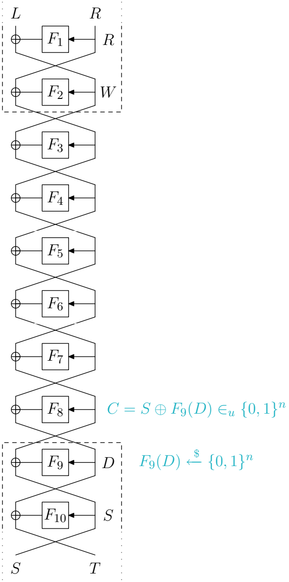
the simulator can always adapt

- $F_9(D)$ is defined randomly



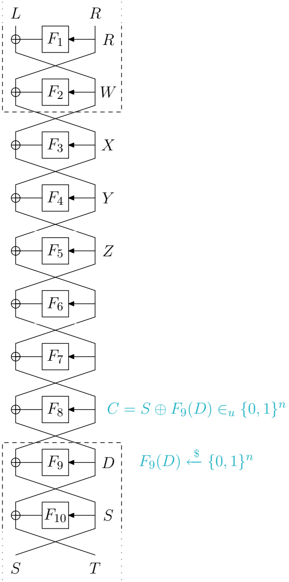
the simulator can always adapt

- $F_9(D)$ is defined randomly
- $\Rightarrow C = S \oplus F_9(D)$ is uniformly distributed and is in the history of F_8 only with negl. prob.



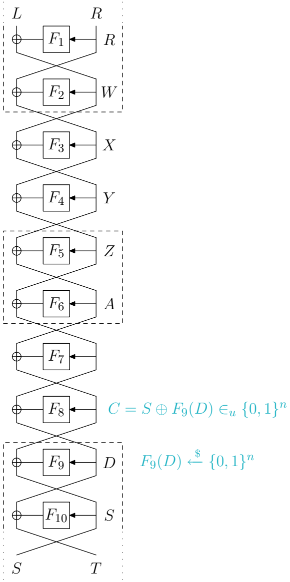
the simulator can always adapt

- $F_9(D)$ is defined randomly
- $\Rightarrow C = S \oplus F_9(D)$ is uniformly distributed and is in the history of F_8 only with negl. prob.



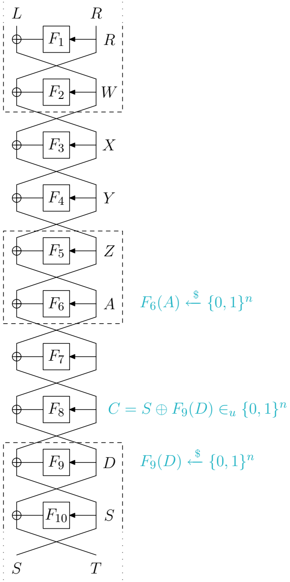
the simulator can always adapt

- $F_9(D)$ is defined randomly
- $\Rightarrow C = S \oplus F_9(D)$ is uniformly distributed and is in the history of F_8 only with negl. prob.
- A cannot be in the history of F_6, otherwise the center (Z, A) would already have been completed



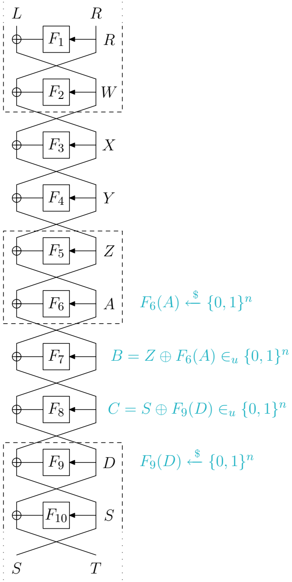
the simulator can always adapt

- $F_9(D)$ is defined randomly
- $\Rightarrow C = S \oplus F_9(D)$ is uniformly distributed and is in the history of F_8 only with negl. prob.
- A cannot be in the history of F_6, otherwise the center (Z, A) would already have been completed
- $\Rightarrow F_6(A)$ is defined randomly

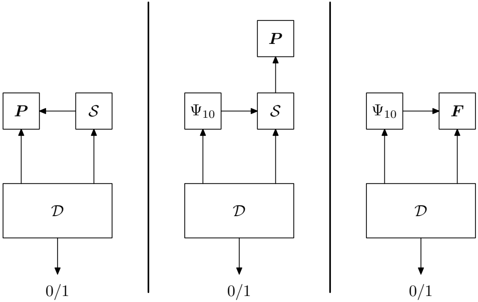


the simulator can always adapt

- $F_9(D)$ is defined randomly
- $\Rightarrow C = S \oplus F_9(D)$ is uniformly distributed and is in the history of F_8 only with negl. prob.
- A cannot be in the history of F_6, otherwise the center (Z, A) would already have been completed
- $\Rightarrow F_6(A)$ is defined randomly
- $\Rightarrow B = Z \oplus F_6(A)$ is uniformly distributed and is in the history of F_7 only with negl. prob.



indistinguishability of the two systems



- *left to middle*: the simulator is consistent with $\boldsymbol{P}$
- *middle to right*: the answers of the simulator are statistically close to random
- conclusion: $\Psi_{10}^{\boldsymbol{F}}$ is indifferentiable from $\boldsymbol{P}$
- for 6 rounds, same ideas plus some subtle technicalities...

applications

- construction of public permutations (*e.g.* for permutation-based hashing or PK encryption)

applications

- construction of public permutations (*e.g.* for permutation-based hashing or PK encryption)
- example of the Phan-Pointcheval 3R-OAEP scheme:
  - in the random permutation model for $\boldsymbol{P}$

$$\mathrm{Enc}_{pk}(m; r) = \mathrm{TOWP}_{pk}(\boldsymbol{P}(m \| r))$$

  - can be replaced in the ROM by a 3R Feistel scheme

$$s = m \oplus \boldsymbol{F}_1(r); \quad t = r \oplus \boldsymbol{F}_2(s); \quad u = s \oplus \boldsymbol{F}_3(t)$$
$$\mathrm{Enc}_{pk}(m; r; \rho) = \mathrm{TOWP}_{pk}(t \| u \| \rho)$$

applications

- construction of public permutations (*e.g.* for permutation-based hashing or PK encryption)
- example of the Phan-Pointcheval 3R-OAEP scheme:
  - in the random permutation model for $\boldsymbol{P}$

$$\mathrm{Enc}_{pk}(m; r) = \mathrm{TOWP}_{pk}(\boldsymbol{P}(m \| r))$$

  - can be replaced in the ROM by a 3R Feistel scheme

$$s = m \oplus \boldsymbol{F}_1(r); \quad t = r \oplus \boldsymbol{F}_2(s); \quad u = s \oplus \boldsymbol{F}_3(t)$$
$$\mathrm{Enc}_{pk}(m; r; \rho) = \mathrm{TOWP}_{pk}(t \| u \| \rho)$$

- example of the Even-Mansour cipher: $E_{k_1,k_2}(m) = k_2 \oplus \boldsymbol{P}(m \oplus k_1)$
  - secure in the random permutation model for $\boldsymbol{P}$
  - secure in the ROM with a 4R Feistel scheme [GentryR04]

applications

- construction of public permutations (*e.g.* for permutation-based hashing or PK encryption)
- example of the Phan-Pointcheval 3R-OAEP scheme:
  - in the random permutation model for $\boldsymbol{P}$

$$\mathrm{Enc}_{pk}(m;r) = \mathrm{TOWP}_{pk}(\boldsymbol{P}(m\|r))$$

  - can be replaced in the ROM by a 3R Feistel scheme

$$s = m \oplus \boldsymbol{F}_1(r); \quad t = r \oplus \boldsymbol{F}_2(s); \quad u = s \oplus \boldsymbol{F}_3(t)$$
$$\mathrm{Enc}_{pk}(m;r;\rho) = \mathrm{TOWP}_{pk}(t\|u\|\rho)$$

- example of the Even-Mansour cipher: $E_{k_1,k_2}(m) = k_2 \oplus \boldsymbol{P}(m \oplus k_1)$
  - secure in the random permutation model for $\boldsymbol{P}$
  - secure in the ROM with a 4R Feistel scheme [GentryR04]
- a dedicated analysis will often enable to replace a random permutation by a Feistel scheme with < 6 rounds

conclusion and open questions

Theorem

The 6-round Luby-Rackoff construction with public random inner functions is indifferentiable from a random permutation.

- the result does not guaranty anything when the internal functions are not perfect

conclusion and open questions

Theorem

The 6-round Luby-Rackoff construction with public random inner functions is indifferentiable from a random permutation.

- the result does not guaranty anything when the internal functions are not perfect
- the result says nothing about the rightfulness to replace an ideal cipher by AES, or a random oracle by SHAx (recent results show this may be risky [BiryukovKN09,LeurentN09])

conclusion and open questions

Theorem

The 6-round Luby-Rackoff construction with public random inner functions is indifferentiable from a random permutation.

- the result does not guaranty anything when the internal functions are not perfect
- the result says nothing about the rightfulness to replace an ideal cipher by AES, or a random oracle by SHAx (recent results show this may be risky [BiryukovKN09,LeurentN09])
- weaker (but still useful) models of indifferentiability: honest-but-curious model [DodisP06], correlation intractability [CanettiGH98]

conclusion and open questions

Theorem

The 6-round Luby-Rackoff construction with public random inner functions is indifferentiable from a random permutation.

- the result does not guaranty anything when the internal functions are not perfect
- the result says nothing about the rightfulness to replace an ideal cipher by AES, or a random oracle by SHAx (recent results show this may be risky [BiryukovKN09,LeurentN09])
- weaker (but still useful) models of indifferentiability: honest-but-curious model [DodisP06], correlation intractability [CanettiGH98]
- open questions:
  - improve the tightness of the analysis, best (exponential) attacks
  - minimal number of calls to the random oracle to build a random permutation: are there constructions with < 6 calls to the RO?

outline

indifferentiability

equivalence of the ROM and the ICM

doubling the domain of an ideal cipher

statement of the problem

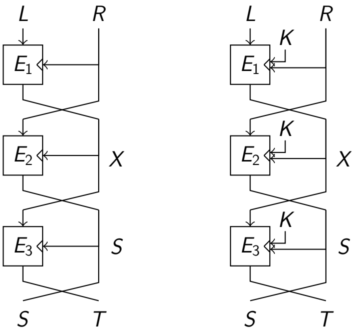
- example of the Phan-Pointcheval 3R-OAEP scheme in the random permutation model for $\boldsymbol{P}$

$$\mathsf{Enc}_{pk}(m; r) = \mathsf{TOWP}_{pk}(\boldsymbol{P}(m\|r))$$

- how to instantiate the permutation $\boldsymbol{P}$ on 1024 or 2048 bits with, say, AES-128?
- previous domain extenders for ciphers (*e.g.* CMC, EME, TET...) were concerned only with conserving pseudorandomness (disk encryption), but they are not indifferentiable from an ideal cipher

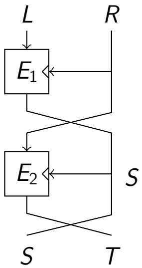
an indifferentiable construction [CoronDMS10]

- this 3R-Feistel-like construction is indifferentiable from a random permutation
- prepending a key K to the 3 ideal ciphers gives a construction indifferentiable from an IC



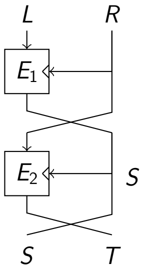
attack against two rounds

- notation: $E(\text{key}, \text{message})$
- $\Psi_2(L \| R) = S \| T$
  with $S = E_1(R, L)$ and $T = E_2(S, R)$
- attack works as follows:



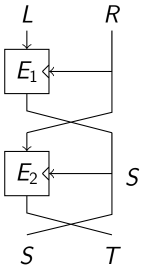
attack against two rounds

- notation: $E(\text{key}, \text{message})$
- $\Psi_2(L\|R) = S\|T$
  with $S = E_1(R, L)$ and $T = E_2(S, R)$
- attack works as follows:
  - choose $R = 0^n$ and $S = 0^n$



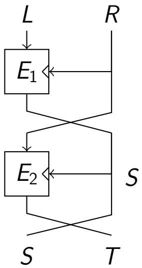
attack against two rounds

- notation: $E(\text{key}, \text{message})$
- $\Psi_2(L \| R) = S \| T$
  with $S = E_1(R, L)$ and $T = E_2(S, R)$
- attack works as follows:
  - choose $R = 0^n$ and $S = 0^n$
  - query $L_0 = E_1^{-1}(R, S)$ and
    $T_0 = E_2(S, R)$



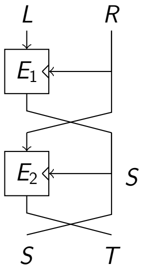
attack against two rounds

- notation: $E(\text{key}, \text{message})$
- $\Psi_2(L \| R) = S \| T$
  with $S = E_1(R, L)$ and $T = E_2(S, R)$
- attack works as follows:
  - choose $R = 0^n$ and $S = 0^n$
  - query $L_0 = E_1^{-1}(R, S)$ and
    $T_0 = E_2(S, R)$
  - then $\Psi_2(L_0, 0^n) = (0^n, T_0)$



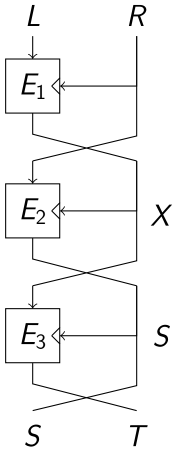
attack against two rounds

- notation: $E(\text{key}, \text{message})$
- $\Psi_2(L\|R) = S\|T$
  with $S = E_1(R, L)$ and $T = E_2(S, R)$
- attack works as follows:
  - choose $R = 0^n$ and $S = 0^n$
  - query $L_0 = E_1^{-1}(R, S)$ and
    $T_0 = E_2(S, R)$
  - then $\Psi_2(L_0, 0^n) = (0^n, T_0)$
- such an I/O pair can be found only with negligible probability for a random permutation



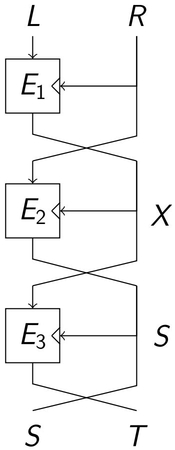
simulation strategy

- on a query $E_1(L, R)$:



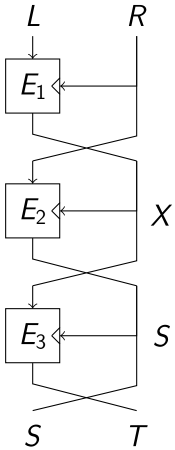
simulation strategy

- on a query $E_1(L, R)$:
  - define $E_1(R, L) \xleftarrow{\text{rand}} X$



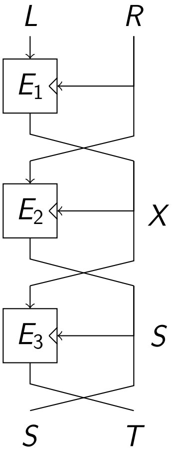
simulation strategy

- on a query $E_1(L, R)$:
  - define $E_1(R, L) \xleftarrow{\text{rand}} X$
  - query $S \| T \leftarrow \boldsymbol{P}(L|R)$



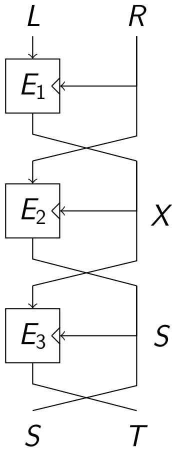
simulation strategy

- on a query $E_1(L, R)$:
  - define $E_1(R, L) \xleftarrow{\text{rand}} X$
  - query $S \| T \leftarrow \boldsymbol{P}(L|R)$
  - set $E_2(X, R) = S$ and $E_3(S, X) = T$ so that $\Psi_3(L\|R) = \boldsymbol{P}(L\|R) = S\|T$



simulation strategy

- on a query $E_1(L, R)$:
  - define $E_1(R, L) \xleftarrow{\text{rand}} X$
  - query $S\|T \leftarrow \boldsymbol{P}(L|R)$
  - set $E_2(X, R) = S$ and $E_3(S, X) = T$ so that $\Psi_3(L\|R) = \boldsymbol{P}(L\|R) = S\|T$
- same strategy for other queries
- the simulator aborts if it cannot define a permutation for some E_i



practical considerations

- extending the key: one can use a random oracle H to define

$$E'(K', M) = E(H(K'), M)$$

practical considerations

- extending the key: one can use a random oracle H to define

$$E'(K', M) = E(H(K'), M)$$

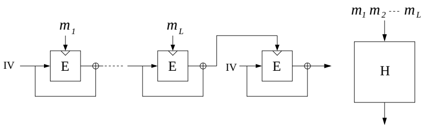
- going beyond double: recursive construction
  - extending the domain by a factor t requires $\mathcal{O}(t^{\log_2(3)}) \simeq \mathcal{O}(t^{1.6})$ applications of the original cipher
  - quickly unpractical

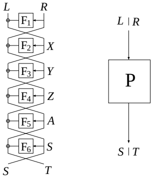
practical considerations

- extending the key: one can use a random oracle H to define

$$E'(K', M) = E(H(K'), M)$$

- going beyond double: recursive construction
  - extending the domain by a factor t requires $\mathcal{O}(t^{\log_2(3)}) \simeq \mathcal{O}(t^{1.6})$ applications of the original cipher
  - quickly unpractical
- alternative construction: build a random oracle with n-bit output from the ideal cipher, and use the 6-round Feistel construction to get a $2n$-bit ideal cipher





thanks for your attention

comments or questions?