Indifferentiable Authenticated Encryption

Manuel Barbosa (Porto)  Pooya Farshim (CNRS & ENS)
Indifferentiable Authenticated Encryption

Manuel Barbosa

Pooya Farshim
(CNRS & ENS)
Hash Functions

long & arbitrary \rightarrow SHA \rightarrow short & random-looking
Hash Functions

long & arbitrary

SHA

short & random-looking

Provably security not always possible.
Random Oracles

long & arbitrary → SHA → short & random-looking
Random Oracles

long & arbitrary → Random Function → short & random-looking
Random Oracles

long & arbitrary → Random Oracle → short & random-looking
Random Oracles are Practical

Provable Security for
Many Simple & Efficient Protocols

- Public-Key Enc. (OAEP, ECIES)
- Signatures (PSS, FDH)
- TLS 1.3
- Symmetric schemes
- ....

Ideal Hash
This Talk
This Talk

Encryption
This Talk

Ideal Hash

Encryption
This Talk

Ideal Hash

Inherit all strengths

Encryption
This Talk

Ideal Hash

Inherit all strengths

Ideal Encryption
This Talk

Ideal Hash

Inherit all strengths

Ideal Encryption

Random Function
This Talk

Ideal Hash

Inherit all strengths

Ideal Encryption

Random Function

What Object?
Authenticated Encryption
Authenticated Encryption

1. $K \leftarrow \text{Gen}(1^\lambda)$
2. $C \leftarrow \text{Enc}(K,N,A,M,\tau)$ \quad $|C| = |M| + \tau$
3. $M/\bot \leftarrow \text{Dec}(K,N,A,C,\tau)$
Authenticated Encryption

1. $K \leftarrow \text{Gen}(1^\lambda)$
2. $C \leftarrow \text{Enc}(K,N,A,M,\tau)$ \hspace{1cm} $|C| = |M| + \tau$
3. $M/\bot \leftarrow \text{Dec}(K,N,A,C,\tau)$

Security says: under an unknown random key $K$
- Nothing about messages leaks
- Cannot forge new valid ciphertexts
Simplifying

1. $K \leftarrow \text{Gen}(1^\lambda)$
2. $C \leftarrow \text{Enc}(K, M) \quad |C| = |M| + \tau$
3. $M/\perp \leftarrow \text{Dec}(K, C)$
Simplifying

1. \( K \leftarrow \text{Gen}(1^\lambda) \)
2. \( C \leftarrow \text{Enc}(K,M) \quad |C| = |M| + \tau \)
3. \( M/\bot \leftarrow \text{Dec}(K,C) \)

A Keyed Injection
Ideal Encryption
Ideal Encryption

Hash
Ideal Encryption

Hash ↔ Function
Ideal Encryption

Ideal Hash ↔ Function
Ideal Encryption

Ideal Hash $\leftrightarrow$ Random Function
Ideal Encryption

Ideal Hash → Random Function
Cipher
Ideal Encryption

Ideal Hash ←→ Random Function
Cipher ←→ Keyed Permutation
Ideal Encryption

Ideal Hash ↔ Random Function
Ideal Cipher ↔ Keyed Permutation
Ideal Encryption

Ideal Hash ↔ Random Function
Ideal Cipher ↔ Random Keyed Permutation
Ideal Encryption

Ideal Hash ↔ Random Function
Ideal Cipher ↔ Random Keyed Permutation
Encryption
Ideal Encryption

Ideal Hash $\leftrightarrow$ Random Function
Ideal Cipher $\leftrightarrow$ Random Keyed Permutation
Encryption $\leftrightarrow$ Keyed Injection
Ideal Encryption

Ideal Hash ↔ Random Function
Ideal Cipher ↔ Random Keyed Permutation
Ideal Encryption ↔ Keyed Injection
Ideal Encryption

Ideal Hash <-> Random Function
Ideal Cipher <-> Random Keyed Permutation
Ideal Encryption <-> Random Keyed Injection
Ideal Encryption

Ideal Hash ↔ Random Function
Ideal Cipher ↔ Random Keyed Permutation
Ideal Encryption ↔ Random Keyed Injection

New Ideal Model
Encryption

Ideal Hash ← Random Function
Inherit all strengths
Ideal Encryption ← What Object?
Encryption

Ideal Hash

Inherit all strengths

Ideal Encryption

Random Function

Random Keyed Injection
Encryption

Ideal Hash

Inherit all strengths

Ideal Encryption

Random Function

Random Keyed Injection
Encryption

- Ideal Hash
  - Inherit all strengths
  - Ideal Encryption

- Random Function

- Indifferentiability

- Random Keyed Injection
Indifferentiability

$C^O$ is "as good as" iEnc:
Indifferentiability

$C^{RO}$ is "as good as" $iEnc$:

$C^{RO} \approx iEnc$
Indifferentiability

$C^\text{RO}$ is "as good as" $i\text{Enc}:

C^\text{RO}, RO \approx i\text{Enc}$
Indifferentiability

$C^\text{RO}$ is “as good as” $i\text{Enc}$:

$C^\text{RO}, RO \approx i\text{Enc}, RO$
Indifferentiability

\( C^\text{RO} \) is "as good as" \( \text{iEnc} \):

\[
C^\text{RO}, RO \approx \text{iEnc}, S^\text{Enc}
\]
Indifferentiability

\( C^{RO} \) is "as good as" \( i\text{Enc} \):

\[ C^{RO}, RO \approx i\text{Enc}, S^{i\text{Enc}} \]
Indifferentiability

$\text{C}^{\text{RO}}$ is “as good as” $\text{iEnc}$:

$\text{C}^{\text{RO}}, \text{RO} \approx \text{iEnc}, \text{S}^{\text{iEnc}}$

Unified Attack Surface
Indifferentiability

$C^R$ is "as good as" iEnc:

$$C^R, RO \approx iEnc, S^{iEnc}$$

Unified Attack Surface

Keys can be under adversarial control
Why Indifferentiability?

**Theorem [MRH04]:** If $C^\text{RO}$ is indifferentiable from $i\text{Enc}$, then it is secure in many adversarial environments in the RO model.
Why Indifferentiability?

**Theorem [MRH04]:** If $C^R$ is indifferentiable from $iEnc$, then it is secure in many adversarial environments in the RO model.

AE, MRAE, & RAE
Why Indifferentiability?

Theorem [MRH04]: If $C^{\text{RO}}$ is indifferentiable from $\text{iEnc}$, then it is secure in many adversarial environments in the RO model.

- AE, MRAE, & RAE
- KDM Security
- Leakage Resilience
- RKA Security
- Committing Encryption
- Deduplication
**Why Indifferentiability?**

*Theorem* [MRH04]: If $C^{RO}$ is indifferentiable from $iEnc$, then it is secure in many adversarial environments in the RO model.

- AE, MRAE, & RAE
- KDM Security
- Leakage Resilience
- RKA Security
- Committing Encryption
- Deduplication
- Combined Models
- Unforeseen Models
Why Indifferentiability?

**Theorem** [MRH04]: If $C^\text{RO}$ is indifferentiable from $i\text{Enc}$, then it is secure in many adversarial environments in the RO model.

- **AE, MRAE, & RAE**
- **KDM Security**
- **Leakage Resilience**
- **RKA Security**
- **Committing Encryption**
- **Deduplication**

**Combined Models**

**Unforeseen Models**

Single stage
So...

Are there any indifferentiable encryption schemes out there?
Fig. 2. The eight “favored” A-schemes. These convert an ivE scheme and a vecMAC into an nAE scheme.

The IV is $F_L(\lfloor N, A \rfloor, M)$ and the tag $T$ is either $T = F_L(\lfloor N, A, M \rfloor)$ or $T = F_L(\lfloor N, A, C \rfloor)$. For this diagram we assume $F_{iv} = F_{tag}$.

We also look at the construction of nAE schemes from an nE scheme and a MAC [19, 20]. While nE schemes are not what practice directly provides—no more than pE schemes are—they are trivial to construct from an ivE scheme, and they mesh well with the nAE target. For this nE + MAC $\rightarrow$ nAE problem we identify 20 candidate schemes, which we call N-schemes. Three of them turn out to be secure, all with tight bounds. The security of one scheme we cannot resolve. The other 16 N-schemes are insecure.

Tidy encryption. Our formalization of ivE, nE, and nAE schemes includes a syntactic requirement, tidiness, that, when combined with the usual correctness requirement, demand that encryption and decryption be inverses of each other. (For an ivE scheme, correctness says that $E_K(\lfloor IV, M \rfloor) \neq \perp$ implies that $D_K(\lfloor IV, C \rfloor) = M$, while tidiness says that $D_K(\lfloor IV, C \rfloor) = M \neq \perp$ implies that $E_K(\lfloor IV, M \rfloor) = C$.) In the context of deterministic symmetric encryption, we regard sloppy schemes—those that are not tidy—as perilous in practice, and needlessly degenerate. Tidiness, we feel, is what one should demand.

A preemptive warning against misinterpretation. A body of results (eg, [8, 22]) have shown traditional MAC-then-Encrypt (MtE) schemes to be difficult to use properly in practice. Although some of our secure schemes can be viewed as being in the style of MtE, the results of this paper should not be interpreted as providing blanket support for MtE schemes. We urge extreme caution when applying any generic composition result from the literature, as implementers
Generic Composition \[\text{[NRS14]}\]

Reconsidering Generic Composition

Fig. 2. The eight “favored” A-schemes.

These convert an ivE scheme and a vecMAC into an nAE scheme.

The IV is \(F_L(\text{[N,A][M]})\) and the tag \(T\) is either \(T = F_L(\text{N,A,M})\) or \(T = F_L(\text{N,A,C})\). For this diagram we assume \(\text{iv}_\text{iv} = \text{tag}\).

We also look at the construction of nAE schemes from an nE scheme and a MAC \[\text{[19, 20]}\].

While nE schemes are not what practice directly provides—no more than pE schemes are—they are trivial to construct from an ivE scheme, and they mesh well with the nAE target. For this nE + MAC → nAE problem we identify 20 candidate schemes, which we call N-schemes. Three of them turn out to be secure, all with tight bounds. The security of one scheme we cannot resolve. The other 16 N-schemes are insecure.

Tidy encryption.

Our formalization of ivE, nE, and nAE schemes includes a syntactic requirement, \text{tidiness}, that, when combined with the usual correctness requirement, demands that encryption and decryption be inverses of each other. (For an ivE scheme, correctness says that \(E_K(\text{IV},M) = C \neq \perp\) implies that \(D_K(\text{IV},C) = M\), while tidiness says that \(D_K(\text{IV},C) = M \neq \perp\) implies that \(E_K(\text{IV},M) = C\).)

In the context of deterministic symmetric encryption, we regard sloppy schemes—those that are not tidy—as perilous in practice, and needlessly degenerate. Tidiness, we feel, is what one should demand.

Were sloppy nE and ivE schemes allowed, the generic composition story would shift again: only schemes A5 and A6, B5 and B6, and N2 would be generically secure. The sensitivity of GC to the sloppy/tidy distinction is another manifestation of the sensitivity of GC results to definitional choices.

A preemptive warning against misinterpretation.

A body of results (eg, \[\text{[8, 22]}\]) have shown traditional MAC-then-Encrypt (MtE) schemes to be difficult to use properly in practice.

Although some of our secure schemes can be viewed as being in the style of MtE, the results of this paper should not be interpreted as providing blanket support for MtE schemes. We urge extreme caution when applying any generic composition result from the literature, as implementers Enc-then-Mac
Reconsidering Generic Composition [NRS14]

Fig. 2. The eight "favored" A-schemes.

These convert an ivE scheme and a vecMAC into an nAE scheme.

The IV is \( \mathcal{F}(N,[A,[M]]) \) and the tag \( T \) is either \( T = \mathcal{F}(N,A,M) \) or \( T = \mathcal{F}(N,A,C) \). For this diagram we assume \( \mathcal{F}_{iv} = \mathcal{F}_{tag} \).

We also look at the construction of nAE schemes from an nE scheme and a MAC \([19, 20]\).

While nE schemes are not what practice directly provides—no more than pE schemes are—they are trivial to construct from an ivE scheme, and they mesh well with the nAE target. For this \( nE + MAC \rightarrow nAE \) problem we identify 20 candidate schemes, which we call N-schemes. Three of them turn out to be secure, all with tight bounds. The security of one scheme we cannot resolve. The other 16 N-schemes are insecure.

Tidy encryption.

Our formalization of ivE, nE, and nAE schemes includes a syntactic requirement, tidiness, that, when combined with the usual correctness requirement, demands that encryption and decryption be inverses of each other. (For an ivE scheme, correctness says that \( E_K(IV,M) \neq \perp \) implies that \( D_K(IV,C) = M \), while tidiness says that \( D_K(IV,C) = M \neq \perp \) implies that \( E_K(IV,M) = C \).) In the context of deterministic symmetric encryption, we regard sloppy schemes—those that are not tidy—as perilous in practice, and needlessly degenerate. Tidiness, we feel, is what one should demand.

Were sloppy nE and ivE schemes allowed, the generic composition story would shift again: only schemes A5 and A6, B5 and B6, and N2 would be generically secure. The sensitivity of GC to the sloppy/tidy distinction is another manifestation of the sensitivity of GC results to definitional choices.

A preemptive warning against misinterpretation. A body of results (eg, \([8, 22]\)) have shown traditional MAC-then-Encrypt (MtE) schemes to be difficult to use properly in practice. Although some of our secure schemes can be viewed as being in the style of MtE, the results of this paper should not be interpreted as providing blanket support for MtE schemes. We urge extreme caution when applying any generic composition result from the literature, as implementers...
Generic Composition [NRS14]

Fig. 2. The eight “favored” A-schemes.

These convert an ivE scheme and a vecMAC into an nAE scheme.

The IV is \( F_L(\text{IV} [\text{A}, \text{M}]) \) and the tag \( T \) is either \( T = F_L(\text{IV}, \text{A}, \text{M}) \) or \( T = F_L(\text{IV}, \text{A}, \text{C}) \). For this diagram we assume \( F_{\text{iv}} = F_{\text{tag}} \).

We also look at the construction of nAE schemes from an nE scheme and a MAC [19, 20]. While nE schemes are not what practice directly provides—no more than pE schemes are—they are trivial to construct from an ivE scheme, and they mesh well with the nAE target. For this nE + MAC → nAE problem we identify 20 candidate schemes, which we call N-schemes. Three of them turn out to be secure, all with tight bounds. The security of one scheme we cannot resolve. The other 16 N-schemes are insecure.

Tidy encryption.

Our formalization of ivE, nE, and nAE schemes includes a syntactic requirement, tidiness, that, when combined with the usual correctness requirement, demands that encryption and decryption be inverses of each other. (For an ivE scheme, correctness says that \( E_K(\text{IV}, \text{M}) = \# \neq \bot \) implies that \( D_K(\text{IV}, \text{C}) = \text{M} \), while tidiness says that \( D_K(\text{IV}, \text{C}) = \# \neq \bot \) implies that \( E_K(\text{IV}, \text{M}) = \text{C} \).) In the context of deterministic symmetric encryption, we regard sloppy schemes—those that are not tidy—as perilous in practice, and needlessly degenerate. Tidiness, we feel, is what one should demand.

Were sloppy nE and ivE schemes allowed, the generic composition story would shift again: only schemes A5 and A6, B5 and B6, and N2 would be generically secure. The sensitivity of GC to the sloppy/tidy distinction is another manifestation of the sensitivity of GC results to definitional choices.

A preemptive warning against misinterpretation. A body of results (eg, [8, 22]) have shown traditional MAC-then-Encrypt (MtE) schemes to be difficult to use properly in practice. Although some of our secure schemes can be viewed as being in the style of MtE, the results of this paper should not be interpreted as providing blanket support for MtE schemes. We urge extreme caution when applying any generic composition result from the literature, as implementers
Attack on Enc-then-Mac
**Attack on Enc-then-Mac**

**Construction:** Changing $K$ does not affect $T$

**iEnc:** Random Injection: changing $K$ will change $T$
Attack on Enc-then-Mac

**Construction:** Changing $K$ does not affect $T$

**iEnc:** Random Injection: changing $K$ will change $T$

**Interpretation:** Related-Key Attacks
General Attacks

<table>
<thead>
<tr>
<th>Algo. $\mathcal{A}\mathcal{E}(K, N, A, M, \tau)$</th>
<th>Algo. $\mathcal{A}\mathcal{D}(K, N, A, C, \tau)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(est_0, est_1) \leftarrow \mathcal{I}_e(K, N, A, M, \tau)$</td>
<td>$(dst_0, dst_1) \leftarrow \mathcal{I}_d(K, N, A, C, \tau)$</td>
</tr>
<tr>
<td>$(K', N', M', \tau') \leftarrow \mathcal{E}^h_0(est_0)$</td>
<td>$(K', N', C', \tau') \leftarrow \mathcal{D}^h_0(dst_0)$</td>
</tr>
<tr>
<td>$C' \leftarrow \mathcal{E}(K', N', \varepsilon, M', \tau')$</td>
<td>$M' \leftarrow \mathcal{D}(K', N', \varepsilon, C', \tau')$</td>
</tr>
<tr>
<td>$C \leftarrow \mathcal{E}^h_1(C', est_1)$</td>
<td>$M \leftarrow \mathcal{D}^h_1(M', dst_1)$</td>
</tr>
<tr>
<td>return $C$</td>
<td>return $M$</td>
</tr>
</tbody>
</table>

A template for generic composition.

Two types of attacks based on how information flows.
Attacks: Specifics Schemes

OCB [Rog et al.]

Deoxys [JNP15]

AEZ [HKR17]
So...

Any indifferentiable encryption schemes?
Feistel
Feistel

- \( L \)
- \( R \)
- \( RO_1 \)
- \( RO_2 \)
- \( RO_3 \)
Feistel

RO₁ \rightarrow RO₂ \rightarrow RO₃

Permutation
Feistel

Permutation
Feistel

Permutation
Feistel

Permutation
Feistel

Permutation
**Theorem:** 3 rounds of Feistel are necessary and sufficient to build a random injection.
Feistel

**Theorem:** 3 rounds of Feistel are necessary and sufficient to build a random injection.
Online Encryption

Some applications need to process messages “on the fly”:

Leads to: A definition of online random keyed injections.
Turn It On

**Theorem:** Chaining an AEAD up indifferienciaility turns it on.
Theorem: Chaining an AEAD up indifferentiability turns it on.
Efficiency Lower Bound

**Theorem:** Any indifferentiable construction of a $w_n$-bit random injection from an $n$-bit permutation must place at least $2w-2$ queries.
Efficiency Lower Bound

**Theorem:** Any indifferentiable construction of a $w$-bit random injection from an $n$-bit permutation must place at least $2^w - 2$ queries.

Lower/upper bounds remain open for:

$$\rho := \frac{\# \text{ blocks}}{\# \text{ queries}} \in [2, 3).$$
**Summary**

New view of AE as a random keyed injection; Implies security in many adversarial environments.
Summary

New view of AE as a random keyed injection;
Implies security in many adversarial environments.

Thank you.