OMD version 2.0

Reza Reyhanitabar

Design Team:
Simon Cogliani, Diana Maimut, David Naccache (ENS, France)
Rodrigo Portella do Canto (Paris II - Panthéon-Assas University, France)
Reza Reyhanitabar, Serge Vaudenay, Damian Vizár (EPFL, Switzerland)
CAESAR candidate OMD

- OMD stands for Offset Merkle–Damgård

Compression function-based mode of operation for AEAED

- **Notable Features:**
  - High security level
    - 127-bit security using sha-256
    - 255-bit security using sha-512
  - Provable security (based on a well-studied standard security assumption)
    If the compression function keyed via its message input is PRF then OMD is a secure AEAD.
**Nonce-based** Authenticated Encryption with Associated Data

\[ N: \text{Nonce} \quad (\text{public message number}) \]
\[ M: \text{Plaintext} \quad \text{that needs to be encrypted and authenticated} \]
\[ AD: \text{Associated data} \quad \text{that needs to be authenticated, but must not be encrypted} \]
\[ C: \text{Ciphertext} \]
\[ K: \text{Secret Key} \]
The Security Goal(s)

\[
\text{Adv}_{\Pi}^{\text{priv}}(A) = \Pr[A^{\text{Enc}_K(.,.,.,.)} \Rightarrow 1] - \Pr[A^{\$ (.,.,.,.)} \Rightarrow 1]
\]
The Security Goal(s)

\[
\text{Adv}_{\Pi}^{\text{priv}}(A) = \Pr[A^{Enc_K(\ldots)} \Rightarrow 1] - \Pr[A^{\$}(\ldots) \Rightarrow 1]
\]

\[
\text{Adv}_{\Pi}^{\text{auth}}(A) = \Pr[A^{Enc_K(\ldots), Dec_K(\ldots)} \text{ forges}]
\]

A forges if: \(\exists (N, AD, C) \text{ such that } Dec_K(N, AD, C) \neq \bot \text{ AND no previous query } Enc_K(N, AD, M) \text{ returned } C\)
Assumption: the keyed compression function $F$ is a PRF

MD Preserves PRF (Bellare and Ristenpart, ICALP 2007)
OMD: Making a nonce-based AE Scheme based on the MD construction

Encrypting a message whose length is a multiple of the block length
OMD: Making a nonce-based AE Scheme based on the MD construction

Encrypting a message whose length is a multiple of the block length
OMD: Making a nonce-based AE Scheme based on the MD construction

Encrypting a message whose length is not a multiple of the block length
Handling Associated Data in OMD

when the length of the data is a multiple of the input length.
Handling Associated Data in OMD
when the length of the data is not a multiple of the input length.
OMD

A Secure Nonce-based AE Algorithm that integrates a modified MD pass with XOR MAC
Computing the Masking Sequence: OMD version 1

\[ \Delta_{0,0}^{K,N} = F_K(N \mathbin{||} 10^{n-1-|N|}, 0^m), \quad \Delta_{0,0}^K = 0^n \]

\[ L_* = F_K(0^n, 0^m) \]

\[ L(0) = 4 \cdot L_* \]

for \( i \geq 1 \)

\[ L(i) = 2 \cdot L(i-1) \]

for \( i \geq 1: \)

\[ \Delta_{i,0}^{K,N} = \Delta_{i-1,0}^{K,N} \oplus L(\text{ntz}(i)) \]

\[ \Delta_{i,1}^{K,N} = \Delta_{i,0}^{K,N} \oplus 2 \cdot L_* \]

\[ \Delta_{i,2}^{K,N} = \Delta_{i,0}^{K,N} \oplus 2 \cdot L_* \]

\[ \Delta_{i,0}^K = \Delta_{i-1,0}^K \oplus L(\text{ntz}(i)) \]

for \( i \geq 0: \)

\[ \Delta_{i,1}^{K} = \Delta_{i,0}^{K} \oplus L_* \]
Computing the Masking Sequence: OMD version 2

\[ \Delta_{0,0}^{K,N} = F_K(N \parallel 10^{n-1-|N|}, 0^m), \quad \overline{\Delta}_{0,0}^K = 0^n \]

\[ L_* = F_K(0^n, <\tau>_m) \]

\[ L(0) = 4 \cdot L_* \]

for \( i \geq 1 \)

\[ L(i) = 2 \cdot L(i - 1) \]

\[ \Delta_{i,0}^{K,N} = \Delta_{i-1,0}^{K,N} \oplus L(ntz(i)) \]

\[ \Delta_{i,1}^{K,N} = \Delta_{i,0}^{K,N} \oplus 2 \cdot L_* \]

\[ \Delta_{i,2}^{K,N} = \Delta_{i,0}^{K,N} \oplus 2 \cdot L_* \]

\[ \overline{\Delta}_{i,0}^K = \overline{\Delta}_{i-1,0}^K \oplus L(ntz(i)) \]

\[ \overline{\Delta}_{i,1}^K = \overline{\Delta}_{i,0}^K \oplus L_* \]

\[ \overline{\Delta}_{i,2}^K = \overline{\Delta}_{i,0}^K \oplus L_* \]

\[ \overline{\Delta}_{i,0}^K = \overline{\Delta}_{i-1,0}^K \oplus L(ntz(i)) \]
Security

\[
\text{Adv}^{\text{priv}}_{\text{p-OMD}[F,\tau]}(t, \sigma_e, \ell_{\text{max}}) = \text{Adv}^{\text{prf}}_{F}(t', 2\sigma_e) + \frac{3\sigma_e^2}{2^n}
\]

\[
\text{Adv}^{\text{auth}}_{\text{p-OMD}[F,\tau]}(t, q_v, \sigma, \ell_{\text{max}}) = \text{Adv}^{\text{prf}}_{F}(t', 2\sigma) + \frac{3\sigma^2}{2^n} + \frac{q_v\ell_{\text{max}}}{2^n} + \frac{q_v}{2^\tau}
\]

\(\sigma_e\): total number of calls to the compression function in encryption queries

\(\sigma\): total number of calls to the compression function in all (encryption and verification) queries

\(q_v\): the number of decryption (verification) queries

\(\ell_{\text{max}}\): the maximum number of internal calls to the compression function in any query

\(n\): the output length of the compression function in bits

\(\tau\): the tag length

\(t' = t + cn\sigma\)
We used the XE method to make a tweakable-PRF out of a PRF. This is the cause of the birthday-type term \( \left( \frac{3\sigma^2}{2^n} \right) \) in the security bounds.
We used the XE method to make a tweak able-PRF out of a PRF.

This is the cause of the birthday-type term \( \left( \frac{3\sigma^2}{2^n} \right) \) in the security bounds

Using a native tweakable-and-keyed compression function (i.e. one with dedicated tweak and key inputs) this term can be avoided.

Any such functions?

- Allocating some part of the input for tweak may be an option but this limits the parameters’ sizes compared to the original OMD

- The TWEAKEY Framework (Jean, Nikolić, and Peyrin, ASIACRYPT 2014) seems a promising way toward designing an efficient TWEAKEY Compression Function.
Conclusion

- OMD v2 is a new version of OMD with a minor tweak.
- OMD v2 has the same performance as OMD v1.
- OMD v2 has the same security bounds as OMD v1.
- OMD v2 is NOT susceptible to the tag-length variation misusing attacks, posted on the CAESAR mailing list on 25 April 2014 by the Ascon team.
Thanks!

Questions?