The XCBC-XOR, XECB-XOR and XECB-MAC Modes

Virgil D. Gligor          Pompiliu Donescu

VDG Inc
6009 Brookside Drive
Chevy Chase, Maryland 20815

{gligor, pompiliu}@eng.umd.edu

August 24, 2001
Outline

1. Security Claims

2. Operational Claims

3. Examples: XCBC-XOR, XECB-XOR, XECB-MAC modes

4. Conclusions
1. Security Claims for Authenticated Encryption

1. Security *Claim* = a security *notion* supported by a mode or scheme of encryption

2. Secrecy *Notion* = < Indistinguishability, adaptive Chosen Plaintext Attacks>

3. Integrity (Authenticity) *Notion* = < Existential Forgery protection, (adaptive) Chosen Plaintext Attacks>
2. Operational Claims for Modes of Encryption

*Operational Notion* = < operational goal, mode characteristics >

Operational *Goals*: cost-performance, simplicity, usability

- **cost-performance**:
  - power consumption
  - speed (no. of block-cipher invocations, latency)
  - implementation cost (e.g., hardware “real-estate”)

- **simplicity**
  - single key
  - specifications (e.g., simple operations)
  - same basic structure for
    - authenticated encryption
    - ciphertext authentication (two-pass, two keys)
    - plaintext authentication (MAC)

- **usability in different environments**
  - various keying-state protection mechanisms needed
  - availability of random number generators,
  - error recovery (ECC, no recovery vs. partial recovery)
Operational Characteristics of Modes

**State**: stateless, stateful-sender, stateful

**Degree of parallelism**
- none (sequential)
- interleaved (known/negotiated no. of processing units for parallel operation)
- architecture-independent
  - independent of no. of processing units
  - same overhead for parallel, pipelined or sequential operation
  - out-of-order processing, incremental

**Error recovery**
- interleaving provides some support on a per-segment basis

**Separation of Confidentiality and Integrity protection (e.g., two-pass, two keys)**

**Padding**
- avoid added block-cipher invocation if message length is a multiple no. of blocks.
- avoid added latency (1 block-cipher invocation) caused by “ciphertext stealing”
Examples of State Characteristics of a Mode

Stateless
- needs good, secure source of randomness \textit{per message}
- no state to maintain across messages (other than key)
- Execution: \( \geq n + 3 \) block cipher invocations;
- Latency: \( \geq 2 \) block cipher invocations in parallel execution

Stateful Sender
- state (e.g., message counter) maintained by sender
- protection of sender state (e.g., counter integrity) across messages
- Execution: \( n + 2 \) block cipher invocations
- Latency: 2 block cipher invocations in parallel execution

Stateful
- state: shared variables (other than key)
- protection of state secrecy, integrity across messages
- more susceptible to failures, intrusion
- Execution: \( n + 1 \) block cipher invocations
- Latency: \( = I \) block cipher invocation in parallel execution
3. Authenticated Encryption Modes

1. **IND-CPA secure mode**: processes block $x_i$, $1 \leq i \leq n+1$, and inputs result to block cipher $F_K$
2. “op” has an inverse “op$^{-1}$”
3. Elements $E_i$ are unpredictable, and $1 \leq i \leq n+1$,
   
   $E_{p_i} \cdot \text{op}^{-1} \cdot E_{q_j}$ are unpredictable, where $(p, i) \neq (q, j)$ and messages $p, q$ are encrypted with same key $K$. 

**Motivation for Confidentiality-Centric View**

**Sender Initialization**

\[ z = z_1 \rightarrow z_2 \rightarrow z_3 \rightarrow z_4 \]

**IND-CPA Secure Mode**

\[ \cdots \rightarrow F_K \rightarrow \cdots \]

\[ w_1 \rightarrow w_2 \rightarrow w_3 \rightarrow w_4 \]

**Receiver Initialization**

\[ z_1 \rightarrow z_2 \rightarrow z_3 \rightarrow z_4 \rightarrow w \]

**Observation (1998):** secure message authentication modes (in chosen message attacks) can be obtained from certain IND-CPA secure modes

**Implication:** same mode structure can be used for

1. authenticated encryption (one pass, single cryptographic primitive)
2. ciphertext authentication (two-pass, single cryptographic primitive, two-key separation of confidentiality and integrity)
3. plaintext authentication (MAC)

**Implementation:** with only little added control logic we get (1), (2) and (3)
Examples of Mode Initialization

stateless mode

Random-number generator

$\rightarrow r_0$

$\rightarrow Ind-CPA mode$

$\rightarrow E_i = i \times r_0$

$\rightarrow F_K$

$\rightarrow K$

$\rightarrow \text{sender}$

$\rightarrow y_0$

$\rightarrow \text{receiver}$

stateful-sender mode

$\rightarrow ctr$

$\rightarrow F_K$

$\rightarrow ctr' = ctr + 1$

$\rightarrow r_0$

$\rightarrow Ind-CPA mode$

$\rightarrow E_i = i \times r_0$

$\rightarrow \text{sender}$

$\rightarrow ctr$

$\rightarrow R^*, R$

$\rightarrow E_i = ctr \times R^* + i \times R$

$\rightarrow 1 \leq i \leq n + 1$

$\rightarrow E_{n+1}^* = ctr \times R^*$

$\rightarrow \text{receiver}$

stateful mode

$\rightarrow ctr$

$\rightarrow R^*, R$

$\rightarrow E_i = ctr \times R^* + i \times R$

$\rightarrow 1 \leq i \leq n + 1$

$\rightarrow E_{n+1}^* = ctr \times R^*$

$\rightarrow \text{receiver}$

$\rightarrow R^*, R = \text{random, uniformly distributed, independent l-bit variables}$
Stateless XCBC-XOR

\[
x = x_1 \oplus x_2 \oplus x_3 \oplus x_4
\]

\[
y = y_0 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5
\]

Stateful-Sender (e.g., \( r_0 = F_K(\text{ctr}) \)) and Stateful (e.g., per-key shared VI, \( z_0 = r_0 + \text{VI} \)) XCBC-XOR are also defined

Stateless Performance: > \( n+3 \) blk. cipher invocations, > 2 blk. cipher invocations latency
Stateful-Sender Performance: \( n+3 \) blk. cipher invocations, 2 blk. cipher invocations latency
Stateful Performance: \( n+2 \) blk. cipher invocations, 2 blk. cipher invocations latency

\[
MDC = \bigoplus, E_i = r_0 \times i, \oplus = +, \text{ and others; } \text{Padding} = 10^* \text{ pattern, use } z_0 \text{ if padding is needed, } \neg z_0 \text{ if padding is not needed}
\]
Stateful XECB-XOR

E_i = \text{ctr} \times \text{R}^* + i \times \text{R} \quad 1 \leq i \leq n+1
E_{n+1}^* = \text{ctr} \times \text{Z}^*

Padding
- \text{Z}^* = -\text{R}^* \text{ if unpadded}
- \text{Z}^* = \text{R}^* \text{ if padded}
- padding pattern: 10^*

Performance Optimized
- n+1 block-cipher invocations
- \approx1 block-cipher latency
Stateless XECB-XOR

Stateless Performance: > n+2 blk. cipher invocations, > 2 blk. cipher invocations latency

\[ MDC = \oplus, \quad E_i = r_0 \times i, \quad E_{n+1}^* = (n+2) \times Z^*, \quad \text{op} = +, \text{ and others; } Z^* = -r_0 \text{ or } r_0 \text{ depending upon padding} \]
Stateful-Sender XECB-XOR

Stateful-Sender Performance: \( n+2 \) blk. cipher invocations, \( 2 \) blk. cipher invocations latency

\[
\text{MDC} = \oplus, \quad E_i = r_0 \times i, \quad E_{n+1}^* = (n+2) \times Z^*, \quad \text{op} = +, \quad \text{and others}; \quad Z^* = \neg r_0 \text{ or } r_0 \text{ depending upon padding}
\]
SEGMENTED Stateful-SenderMode*

\( x = x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12} \)

\[ \begin{align*}
\text{Plaintext Segment 1 Encryption} & \rightarrow F_K \rightarrow K \\
\text{Plaintext Segment 2 Encryption} & \rightarrow F_K \rightarrow K \\
\text{Plaintext Segment 3 Encryption} & \rightarrow F_K \rightarrow K
\end{align*} \]

\[ \begin{align*}
\text{Ciphertext Segment 1} & \rightarrow y_1 \ y_2 \ y_3 \ y_4 \ y_5 \\
\text{Ciphertext Segment 2} & \rightarrow y_5 \ y_6 \ y_7 \ y_8 \ y_9 \\
\text{Ciphertext Segment 3} & \rightarrow y_9 \ y_{10} \ y_{11} \ y_{12} \ y_{13}
\end{align*} \]

\[ \begin{align*}
\text{ctr} \quad y = y_1 \ y_2 \ y_3 \ y_4 \ y_5 \ y_6 \ y_7 \ y_8 \ y_9 \ y_{10} \ y_{11} \ y_{12} \ y_{13} \\
\text{ctr}' = \text{ctr}+3
\end{align*} \]

(*) per-segment error handling => error recovery
Stateful XECB-MAC

E_i = ctr x Z^* + i x R, 1 ≤ i ≤ n+1

Padding
- Z^* = ¬R^* if unpadded
- Z^* = R^* if padded
- padding pattern: 10*

Performance Optimized
- n+1 block-cipher invocations
- ≈1 block-cipher latency
4. Conclusions

• **Cost-performance:**
  - stateful XECB-XOR and XECB-MAC are optimal (minimum block cipher invocations and latency);
  - XECB-XOR and XECB-MAC modes exhibit architecture-independent parallelism;
  - XCBC-XOR modes are simple extensions of the standard CBC mode

• **Simplicity:**
  - same basic structure for
    - authenticated encryption
    - ciphertext authentication (two-pass, two keys)
    - plaintext authentication (MAC)

• **Usability:**
  - stateless, stateful-sender, stateful modes for different environments.

• **Security:**
  - good bounds.