Additional Modes for AsCON

Rhys Weatherley
Southern Storm Software, Pty Ltd
NIST Sixth Lightweight Cryptography Workshop, June 2023
Lots of little cryptography experiments from 1990’s to today

1996: SSL 2.0 and 3.0 implementations for Oracle PowerBrowser

2015 to present: Arduino Cryptography Library

- AES, SHA-1, SHA-2, SHA-3, BLAKE-2, ChaCha, SPECK
- GCM, CTR, EAX, CBC, CFB, OFB, HMAC, HKDF, Poly1305
- Curve25519, Ed25519, P521, NewHope
- Added CAESAR finalists Ascon-128 and ACORN-128 in 2018
- AVR, ARM Cortex, and ESP32 platforms
- https://github.com/rweather/arduinolibs

2019 to 2022: Implementing and benchmarking LWC candidates in rounds 2 and 3

Recently: Ascon Suite and Additional Modes for Ascon
ASCON-cXOF - Customizable Hashing

Customizable XOF mode similar to cSHAKE (NIST SP 800-185)

\[ \text{ASCON-cXOF}(X, L, N, C, a, b, r) \]

- \( X \) - Input string of any length
- \( L \) - An integer representing the desired output length (0 for indefinite)
- \( N \) - Function name string; e.g. “KMAC”, “KDF”, “PRNG”, etc (may be empty)
- \( C \) - Customization string for application-specific variants on \( N \) (may be empty)
- \( a \) - Number of ASCON rounds for initialization and finalization \((1 \leq a \leq 12)\)
- \( b \) - Number of ASCON rounds for absorbing and squeezing \((1 \leq b \leq a)\)
- \( r \) - Rate for absorbing and squeezing \((64 \text{ or } 128)\)

Except for the handling of \( N \) and \( C \), ASCON-cXOF is the same as regular ASCON hashing.
ASCON-cXOF - Pseudocode

\textbf{ASCON-cXOF}(X, L, N, C, a, b, r):

\texttt{State} \leftarrow p^a(\text{Format-First-Block}(L, N, a, b, r))

\textbf{if} \ \text{len}(C) > 0 \ \textbf{then}

\hspace{1em} \texttt{State} \leftarrow \text{Absorb}(\texttt{State}, \text{pad}(C), b, r)

\hspace{1em} \texttt{State} \leftarrow \texttt{State} \oplus 1

\texttt{State} \leftarrow \text{Absorb}(\texttt{State}, \text{pad}(X), b, r)

\texttt{State} \leftarrow p^a(\texttt{State})

\textbf{return} \ \text{Squeeze}(\texttt{State}, L, b, r)
ASCON-cXOF - Visual Structure
ASCON-CXOF - Handling the Function Name

ASCON’s hashing mode already encodes $L$, $a$, $b$, and $r$ in the initial block. We can add the function name $N$ to the initial block:

If $\text{len}(N) > 256$, then set $N \leftarrow \text{ASCON-HASH}(N)$
ASCON-cXOF - Parameterization for Common Uses

- \texttt{ASCON-Hash}(X) = \texttt{ASCON-cXOF}(X, 256, "", ",", 12, 12, 64)
- \texttt{ASCON-HashA}(X) = \texttt{ASCON-cXOF}(X, 256, "", ",", 12, 8, 64)
- \texttt{ASCON-Xof}(X) = \texttt{ASCON-cXOF}(X, 0, "", ",", 12, 12, 64)
- \texttt{ASCON-XofA}(X) = \texttt{ASCON-cXOF}(X, 0, "", ",", 12, 8, 64)
- \texttt{ASCON-Kmac}(K, L, X, C) = \texttt{ASCON-cXOF}(K \| X, L, "KMAC", C, 12, 12, 64)
- \texttt{ASCON-KDF}(K, L, C) = \texttt{ASCON-cXOF}(K, L, "KDF", C, 12, 12, 64)
- \texttt{ASCON-Prng}(Seed, C) = \texttt{ASCON-cXOF}(Seed, 0, "PRNG", C, 12, 12, 64)
- ...
Other things in AsCON Suite

- Drop-in replacements for HMAC, HKDF, and PBKDF2.
- Safely transitioning from squeezing back to absorbing.
- Reseedable PRNG using the SpongePRNG construction.
- ...

And obviously:

- AsCON-128, AsCON-128a, AsCON-80pq
- AsCON-HASH, AsCON-HASHA, AsCON-XOF, AsCON-XOFA
- AsCON-Mac, AsCON-Prf, AsCON-PrfShort
- ISAP-A-128, ISAP-A-128a
Ascon as a tweakable block cipher (yes, really)

- Tweakable block ciphers are required for memory and disk encryption.
- On-the-fly memory encryption (using tweaked versions of AES) is increasingly common in microcontrollers.

- AEAD modes are unsuitable because nonce reuse is fatal.
- SIV modes are suitable only if there is extra storage for the tag.
Luby-Rackoff is a method to turn a set of pseudorandom functions $F_i$ into a Feistel block cipher.

- Break the 128-bit input block up into $L$ and $R$ halves.
- For each round, $L \leftarrow L \oplus F_i(R)$
- Swap the two halves in every round except the last.
- 10 rounds are enough for everyone!
- For ASCON: $F_i(R) = \lfloor p^a(R || K_i || T || r_c_i) \rfloor_{64}$
- Reduced-round versions of $p^a$ to improve performance.
- Or ... just use Skinny-128-384+ instead.
Thank You!

https://github.com/rweather/ascon-suite
https://eprint.iacr.org/2023/391/