Skein
More than just a hash function

Third SHA-3 Candidate Conference
23 March 2012
Washington DC
Skein is Skein-512

• Confusion is common, partially our fault
• Skein has two special-purpose siblings:
  – Skein-256 for extreme memory constraints
  – Skein-1024 for the ultra-high security margin

• But for SHA-3, Skein is Skein-512
  – One hash function for all output sizes
Skein Architecture

• Mix function is 64-bit ARX
• Permutation: relocation of eight 64-bit words
• Threefish: tweakable block cipher
  – Mix + Permutation
  – Simple key schedule
  – 72 rounds, subkey injection every four rounds
  – Tweakeable-cipher design key to speed, security
• Skein chains Threefish with UBI chaining mode
  – Tweakeable mode based on MMO
    – Provable properties
    – Every hashed block is unique
• Variable size output means flexible to use!
  – One function for any size output
The Skein/Threefish Mix

\[ R_{r,i} \]
Four Threefish Rounds
Skein and UBI chaining

Configuration block

\[ G_0 \leftarrow UBI(0, C, T_{cfg}2^{120}) \]

Message

\[ G_1 \leftarrow UBI(G_0, M, T_{msg}2^{120}) \]

Output transform

\[ H \leftarrow UBI(G_1, 0, T_{out}2^{120}) \]
Fastest in Software

• 5.5 cycles/byte on 64-bit reference platform
• 17.4 cycles/byte on 32-bit reference platform
• 4.7 cycles/byte on Itanium
• 15.2 cycles/byte on ARM Cortex A8 (ARMv7)
  – New numbers, best finalist on ARMv7 (iOS, Samsung, etc.)
Fast and Compact in Hardware

• Fast
  – Skein-512 at 32 Gbit/s in 32 nm in 58 k gates
  – (57 Gbit/s if processing two messages in parallel)

• To maximize hardware performance:
  – Use a fast adder, rely on simple control structure, and exploit Threefish's opportunities for pipelining
  – Do not trust your EDA tool to generate an efficient implementation

• Compact design:
  – Small FPGA implementation (At et al.)
  – 132 slices and two memory blocks on Virtex-6 FPGA
  – Threefish block cipher “for free” (support ALL symmetric crypto primitives in a single hw system)
Secure

• Conservative design
  • 2x security margin
  • UBI defends against attacks

• Builds on well-understood primitives

• Easy to understand and analyze
  – Only changes have been better constants

• Formal security arguments for the mode
  – Mathematical proof that a weakness in Skein implies a weakness in Threesfish
  – We know how to analyze block ciphers
Secure — Best Attacks

• Rotation (Khovratovich et al.) attacks fixed with new constant
• Differential attack against 34 rounds of Threefish (Aumasson et al.)
• Biclique attack, pseudo-preimages on Skein512 at 37 rounds with $2^{511.2}$ steps (Khovratovich et al.)

• We believe Skein/Threefish is ready to use
Design Maximizes Diffusion

<table>
<thead>
<tr>
<th>Hash Function</th>
<th>Full Diffusion After</th>
<th>Diffusion Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skein</td>
<td>10 rounds (of 72)</td>
<td>7.2</td>
</tr>
<tr>
<td>SHA-1</td>
<td>30 steps (of 80)</td>
<td>2.7</td>
</tr>
<tr>
<td>SHA-256</td>
<td>14 steps (of 64)</td>
<td>4.6</td>
</tr>
<tr>
<td>SHA-512</td>
<td>18 steps (of 80)</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Full diffusion is number of rounds to propagate a single-bit change to all bits.
Flexible

• Hash functions are the utility functions of crypto
• Skein has formalizations of many common uses:
  • Any output size
    – Simplifies a lot of applications from networks to OAEP
• Extra features:
  – One-pass (zero per-message overhead) MAC
  – KDF, PRNG, stream cipher
  – Tree hash and tree MAC
    • Unlimited throughput through parallelism
    • Random-access hash and MAC
Free Block Cipher

• Threefish is the block cipher at the heart of Skein
  – Free: the security of Skein assumes the security of
    Threefish

• Wide block
  – Solves the birthday bound problems we have with
    128-bit block ciphers

• Tweakable: extra flexibility
  – Tweaks + wide block is good for storage and networks

• Provides a fallback for AES
Implementation

• One implementation for any output size!
• Existing implementations in
  – Python, C, C++, C#, Spark, Atmel AVR, x86, x64, ARM, Java, Ada, Cryptol, FPGA, ASIC and more
  – Parallel tree hashing in Java
• Implementation in Spark adds a formal automated correctness-of-implementation proof
Skein: Fast, Secure, Flexible

- Fastest in software, fast in hardware
- Wider security margin than existing primitives
- Skein is designed for the many ways people use hash functions now
- We don't know what future applications hash functions will have, so the best standard is a flexible one