Twister

Ewan Fleischmann  Christian Forler  Michael Gorski

Bauhaus-Universität Weimar

Sirrix AG security technologies

First SHA-3 Conference

February 25, 2009
Outline

1. Introduction
2. Design
   - Mini-Round
   - Compression Function
   - Checksum (only TWISTER-512)
   - Output Transformation
3. Cryptanalysis
4. Benchmarks
5. Conclusions
Motivation

- Feb 2005: Wang et al. published SHA-1 attack.
- Nov 2007: NIST announced SHA-3 contest.
- Oct 2008: Deadline for SHA-3 candidates submission.
- Dec 2008: NIST announced 51 round one candidates.
Overview

- Paragons: AES and Grindahl.
- Variable output length: 32-512 bit.
- Compression function: 512 bit message input.
- Internal State: $8 \times 8$ state matrix with elements of $G(2^8)$ (i.e. bytes).
Mini-Round

Message-Injection

AddTwistCounter

SubBytes

ShiftRows

MixColumns
Message Injection

XOR 64 bit message block with last row.
Add TwistCounter

- XOR 64 bit TwistCounter with 2nd column.
- Post decrementation.
- Initial value: 0xFFFF:FFFF:FFFF:FFFF.
- Prevents slide attacks.
SubBytes

Update each byte by AES S-box lookups.
Shift Rows

Cyclically rotates the bytes of i-th row by (i-1) positions to the left.
Mix Columns

- Multiplication by a MDS matrix.
- MDS matrix: A 'cyclical rotation' of (02 01 01 05 07 08 06 01).
- High speed diffusion in combination with ShiftRows.
**Twister-256: Compression Function**

**Twister-512: Compression Function**
The checksum $C$ is as the state $S$ a $8 \times 8$ matrix.

- Enlarges the state of Twister-512.
- Checksum update after message injection.
  \[ C(i,\downarrow) = C(i,\downarrow) \oplus C(i+1,\downarrow) \boxplus S(i,\downarrow) \]

- Checksum enters state after message processing.
  (as input for an Twister-256 compression function.)
Returns first column of $S^i \oplus S^{i-1}$. 
**Cryptanalysis**

- Mini-Round is collision free.
- Full diffusion after two Mini-Rounds.
- **Twister** uses well known building blocks.
- *(Too?)* easy to analyze.
Full Diffusion

1. Mini-Round
   - Message-Injection
   - ShiftRows
   - MixColumns

2. Mini-Round
   - Message-Injection
   - ShiftRows
   - MixColumns
32-Bit Benchmarks

Setup

- Hardware: Core2Duo T7300 (2.0 Ghz), 2048 MB RAM.
- Operating System: Debian/GNU Linux (Lenny)
- Compiler: GCC-4.1

Benchmarks

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Cycles per Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHA-256</td>
<td>29.3</td>
</tr>
<tr>
<td>Twister-256</td>
<td>35.8</td>
</tr>
<tr>
<td>SHA-512</td>
<td>55.2</td>
</tr>
<tr>
<td>Twister-512</td>
<td>39.6</td>
</tr>
</tbody>
</table>
64-Bit Benchmarks

**Setup**

- **Hardware:** Core2Duo T7300 (2.0 Ghz), 2048 MB RAM.
- **Operating System:** Debian/GNU Linux (Lenny)
- **Compiler:** GCC-4.3

**Benchmarks**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Cycles per Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHA-256</td>
<td>20.1 cycles per byte</td>
</tr>
<tr>
<td>Twister-256</td>
<td>15.8 cycles per byte</td>
</tr>
<tr>
<td>SHA-512</td>
<td>13.1 cycles per byte</td>
</tr>
<tr>
<td>Twister-512</td>
<td>17.5 cycles per byte</td>
</tr>
</tbody>
</table>
Conclusions for **Twister-256/512**

- Fairly fast hash functions  
  (especially in the 'non-optimal' case)
- Relying on provably secure components
- But: **Twister-512** is harmed (see SHA-3 zoo)
- Easy fix available (smaller Maxi-Rounds)