Adiantum: length-preserving encryption for entry-level processors

Paul Crowley and Eric Biggers

Google LLC

March 28, 2019
Overview

- The problem
- The solution
Section 1

The problem
The problem

- Hardware (eg ARM CE) makes AES fast
- ...but some devices don’t have it
The solution (for TLS)

- RFC7539
  - ChaCha for encryption
  - Poly1305 for authentication
- Much faster
But...

- RFC7539 is an AEAD mode, so $|C| > |P|
  - nonce
  - MAC
- Storage encryption requires $|C| = |P|$
Full disk encryption

- 4KiB virtual sector $\leftrightarrow$ 4KiB real sector
- No special flash hardware
File based encryption

- Databases update sectors
- If read/write of one sector touches two sectors...
  - Atomicity more difficult
  - Speed is halved
  - Lifetime is halved
Android “Compatibility Definition Document”, version 8.1, section 9.9:

If device implementations [...] support data storage encryption with Advanced Encryption Standard (AES) crypto performance above 50MiB/sec, they MUST enable the data storage encryption by default [...]

Section 2

The solution
Formal properties

- Deterministic
- No nonce
- **Tweakable super-pseudorandom permutation (SPRP)**
  - family of permutations indexed by tweak and length
  - indistinguishable from random permutations
  - attacker can query $f, f^{-1}$
AES-XTS

- 128-bit tweakable SPRP
- 4KiB sector: applied 256 times
- Two-part tweak
- Cortex A7: 58.6 cpb (decryption)
Whole sector encryption

- 4KiB tweakable SPRP
- every bit of plaintext affects all of ciphertext
- every bit of ciphertext affects all of plaintext
- every tweak a new permutation
- opportunity to be faster
Three-pass structure

- SPRP: read all before writing any
- same in decryption direction
- minimum three passes
- hash-XOR-hash faster than XOR-hash-XOR
HCTR, HCH

- hash-XOR-hash structure
- Block cipher defeats LR attack
- But no faster on our hardware (AES, $GF(2^{128})$)
H PolyC and Adiantum

- Similar structure: hash-XOR-hash with block cipher
- More parallel decryption
- Use RFC7539 primitives
- H PolyC-ChaCha20-AES: 17.8 cpb
- Use ChaCha12 instead: H PolyC, 13.6 cpb
- Use NH
- ...but combine with Poly1305
- Adiantum: 10.6 cpb
## Performance

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>cbp (4096)</th>
<th>cpb (512)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adiantum-XChaCha12-AES</td>
<td>10.6</td>
<td>15.8</td>
</tr>
<tr>
<td>HPolyC-XChaCha12-AES</td>
<td>13.6</td>
<td>18.7</td>
</tr>
<tr>
<td>Adiantum-XChaCha20-AES</td>
<td>14.7</td>
<td>20.2</td>
</tr>
<tr>
<td>Speck128/256-XTS</td>
<td>15.8</td>
<td>16.9</td>
</tr>
<tr>
<td>HPolyC-XChaCha20-AES</td>
<td>17.8</td>
<td>23.4</td>
</tr>
<tr>
<td>NOEKEON-XTS</td>
<td>26.9</td>
<td>27.9</td>
</tr>
<tr>
<td>AES-128-XTS (decryption)</td>
<td>42.7</td>
<td>43.9</td>
</tr>
<tr>
<td>AES-256-XTS (decryption)</td>
<td>58.6</td>
<td>60.1</td>
</tr>
</tbody>
</table>
Proof (main step)

- Adversary distinguishes world X and world Y
- Plaintext, ciphertext queries, any length and tweak
- World X: Adiantum, with random permutation $\pi$ and random function $f$
- World Y: all answers random
- H-coefficient technique
- After final query, attacker gets the hash key
Bad transcripts
Bad transcripts

- Results are random in world Y
- Collision in result: $2^{-128}$
- We forbid pointless queries
- Collision in query: at most $\epsilon$
- Total across all queries: at most $(\epsilon + 2^{-128})q/2$
Good transcripts

- In world Y, all responses have probability $2^{-|P|}$
- In world X
  - probability $f$ has right output: $2^{-(|P|-128)}$
  - probability $\pi$ has right output: $\frac{1}{2^{128-i}}$
  - where $i$ is the number of queries before this one
- These are independent, so overall probability is $2^{-(|P|-128)} \frac{1}{2^{128-i}}$
- ...which is equal to or slightly larger than $2^{-|P|}$
H-coefficient technique

- Every good transcript is at least as likely in world X as world Y
- Probability of bad transcript $\leq (\epsilon + 2^{-128})q^2$
- By H-coefficient technique, distinguishing advantage $\leq (\epsilon + 2^{-128})q^2$
Security

Distinguishing bound quadratic in queries, linear in message/tweak length

\[
(3(2^{-128}) + 2^{-103} \max(1 + \lceil l_T/128 \rceil, 2 \lceil (l_M - 128)/8192 \rceil))(\frac{q}{2})^2
\]

\[+ \operatorname{Adv}^\text{sc}_{S_{KS}} (1 + q, 9088 + q(l_M - 128), t') + \operatorname{Adv}^\pm_{E_{KE}} (q, t')\]

where

- \( q \): number of queries
- \( l_T, l_M \): maximum length of tweak, message in bits
- \( \operatorname{Adv}^\pm_{E_{KE}} (q, t') \): distinguishing advantage against AES-256
- \( \operatorname{Adv}^\text{sc}_{S_{KS}} (q, l, t') \): distinguishing advantage against XChaCha12
- \( t' = t + \mathcal{O}(q(l_T + l_M)) \)
Adiantum in Android

- Part of Linux 5.0
- Android “dessert” releases
  - Cupcake, Donut, Eclair, ...
- Some Android Pie devices will use it
- No carveout: devices shipping “Q” will all be encrypted