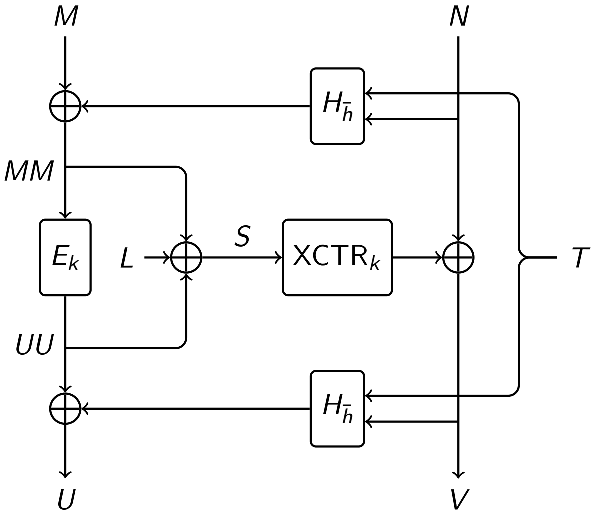
HCTR2

Paul Crowley, Eric Biggers,
Nathan Huckleberry

Google LLC

2023-10-03



Background: Adiantum

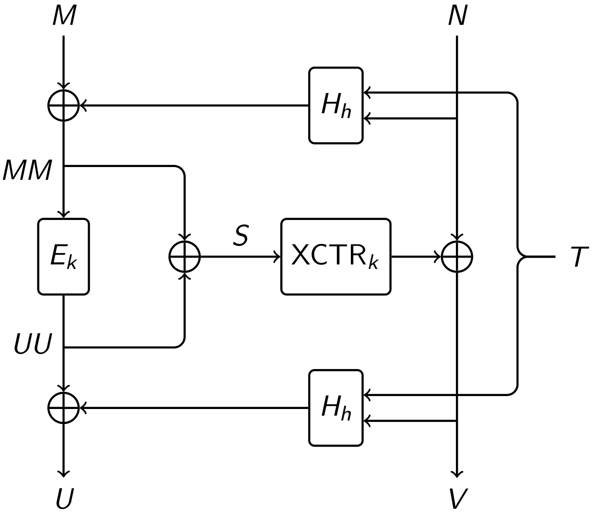
- 2018: *Adiantum: length-preserving encryption for entry-level processors*
- A wide-block mode
- Fast without AES+GHASH instructions
- Efficient on 0.5kB-4kB messages

What we needed

- A wide-block mode
- Fast *with* AES+GHASH instructions
- Efficient on short messages (16B-64B)
- Secure and fully specified

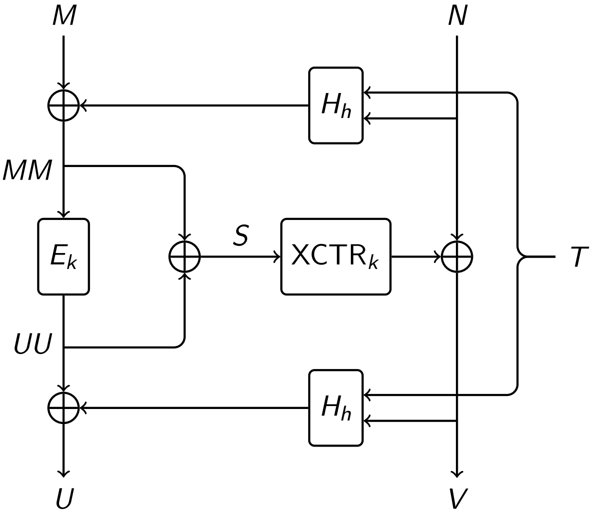
HCTR

- Lots of proposals from around 2005–2009
  - CMC, EME, EME*, PEP, TET, HEH, HCH, HSE, HMC...
- HCTR: Wang, Feng, and Wu 2005
- Quadratic security: Chakraborty and Nandi 2008



HCTR

- Simple
- Hash-encrypt-hash structure
- Fast with AES+GHASH instructions
- No ciphertext stealing
- XCTR mode



XCTR mode

- CTR: nonce PLUS counter
- XCTR: nonce XOR counter
- No 128-bit addition required
- No GCM hack
- Little-endian

$$
\begin{aligned}
\mathrm{CTR}_k(S) =& E_k(\mathrm{bin}(S+1)) \\
&\| E_k(\mathrm{bin}(S+2)) \\
&\| E_k(\mathrm{bin}(S+3)) \| \cdots \\
\mathrm{XCTR}_k(S) =& E_k(S \oplus \mathrm{bin}(1)) \\
&\| E_k(S \oplus \mathrm{bin}(2)) \\
&\| E_k(S \oplus \mathrm{bin}(3)) \| \cdots
\end{aligned}
$$

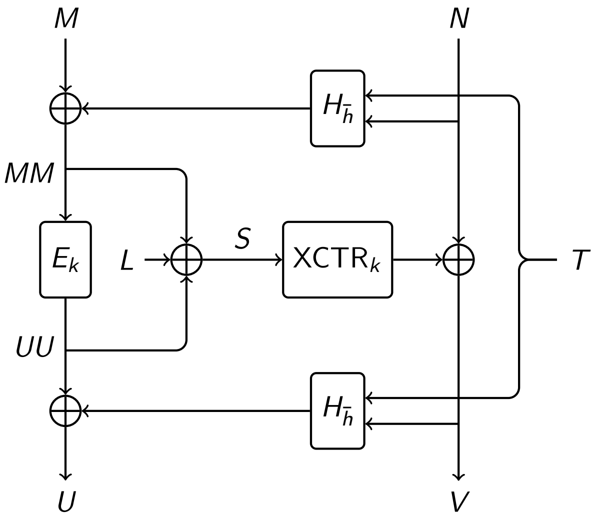
HCTR issues

- Hash encoding is non injective
  - $H_h(0) = h = H_h(\lambda)$
- Error in quadratic security proof
- HCTR2 fixes these, and "sands the edges"

```
procedure Hash(h, T, M)
    return H_h(M‖T)
end procedure
procedure H(h, X)
    if |X| = 0 then
        return h
    else
        return polyeval(h, pad(X)‖ bin(|X|))
    end if
end procedure
```

HCTR2

- New key-dependent constant L XORed into S
- Rescues quadratic security bound



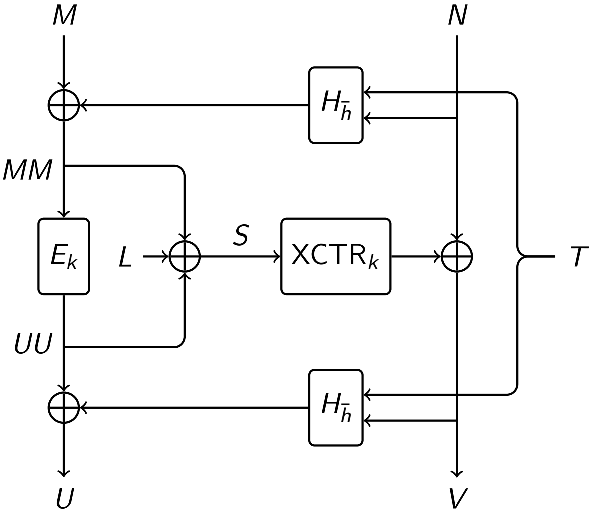
HCTR2 hash function

- Fixes encoding to be injective
- Handles variable-length tweak
- Length+tweak processed only once
- Uses POLYVAL for speed

procedure $\textsc{Hash}(\bar{h}, T, M)$
  if $|M| \bmod n = 0$ **then**
    $X \leftarrow \mathrm{bin}(2|T| + 2) \| \mathrm{pad}(T) \| M$
  else
    $X \leftarrow \mathrm{bin}(2|T| + 3) \| \mathrm{pad}(T) \| \mathrm{pad}(M \| 1)$
  end if
  return $\mathrm{POLYVAL}(\bar{h}, X)$
end procedure

Sanding the edges

- $\bar{h}$, L derived from block cipher
- Endianness, field convention specified
- Sample implementation and test vectors
- In Linux kernel now



Quadratic security

- q queries, σ blocks, t time
- H-coefficient based proof

$$
\begin{aligned}
&\mathsf{Adv}^{\pm\widetilde{\mathrm{prp}}}_{\mathsf{HCTR2}[E]}(q, \sigma, t) \\
\leq\ & \mathsf{Adv}^{\pm\mathrm{prp}}_{E}(\sigma + 2, t + \sigma t') \\
&+ \left(3\sigma^2 + 2q\sigma + q^2 + 7\sigma + 2\right) \Big/ 2^{n+1}
\end{aligned}
$$

Future work: better than quadratic security?

- This is all still speculative
- Inspired by AES-GCM-SIV
- Per-message keys derived from nonce
- Derive $\bar{h}$ and L in the same way
- Multi-target security matters if keys are 128-bit
- Proof in ideal cipher model