Signatures

Watson Ladd

22 June 2015

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it
- Public key $A = aG$, compute $r = H(v||M)$, $R = rG$, $c = H(M||R)$, $s = r - ca$. Send R, s

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it
- Public key $A = aG$, compute $r = H(v||M)$, $R = rG$, $c = H(M||R)$, $s = r - ca$. Send R, s
- Problem: sometimes H isn't long enough

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it
- Public key $A = aG$, compute $r = H(v||M)$, $R = rG$, $c = H(M||R)$, $s = r - ca$. Send R, s
- Problem: sometimes H isn't long enough
- Solution: Use r as the key for HMAC of a counter value

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it
- Public key $A = aG$, compute $r = H(v||M)$, $R = rG$, $c = H(M||R)$, $s = r - ca$. Send R, s
- Problem: sometimes H isn't long enough
- Solution: Use r as the key for HMAC of a counter value
- Or use SHA3, which has arbitrary length output

The proposal redux

- Let G be the generator of an Edwards curve, encoded in the way everyone is doing it
- Public key $A = aG$, compute $r = H(v||M)$, $R = rG$, $c = H(M||R)$, $s = r - ca$. Send R, s
- Problem: sometimes H isn't long enough
- Solution: Use r as the key for HMAC of a counter value
- Or use SHA3, which has arbitrary length output
- Mandate double length scalar: no reason to think harder!

Security

- Assume keys are not used with different hash functions!

Security

- Assume keys are not used with different hash functions!
- Otherwise we have nonstandard, untested assumptons

Security

- Assume keys are not used with different hash functions!
- Otherwise we have nonstandard, untested assumptons
- Under mild assumtions repalce r with random number

Security

- Assume keys are not used with different hash functions!
- Otherwise we have nonstandard, untested assumptons
- Under mild assumtions repalce r with random number
- Apply Forking Lemma

Properties

- Batchable

Properties

- Batchable
- "Online": only need to store two hash states for one pass through M

Properties

- Batchable
- "Online": only need to store two hash states for one pass through M
- Avoids trouble spots for implementors

Properties

- Batchable
- "Online": only need to store two hash states for one pass through M
- Avoids trouble spots for implementors
- Most proposed schemes have this property

Properties

- Batchable
- "Online": only need to store two hash states for one pass through M
- Avoids trouble spots for implementors
- Most proposed schemes have this property
- But most are somewhat more complicated