



Haym @SalomonCrypto

Oct 20 · 25 tweets · [SalomonCrypto/status/1583168818381172736](https://twitter.com/SalomonCrypto/status/1583168818381172736)

Tr

## (1/24) Degen's Handbook: A Practical Guide to Elliptic Curve Pairings

Magic is real, elliptic curve pairings are proof. Through incredibly advanced math, pairings allow us to multiply two polynomials... through elliptic curves!

Here's all you need to know.

### Elliptic Curve Pairings

*Elliptic curve pairings are the elliptic point equivalent of multiplication*

$$e(p, q) = t \approx p * q = t$$

An elliptic curve pairing is a function that takes a pair of elliptic curve points returns an element of some other group, called the target group

Think of a pairing as a black box that takes elliptic points. Pairings cannot be used consecutively; the target group points don't match the input points

Elliptic curve pairings are bilinear, holding to the following property:

$$e(p+r, q) = e(p, q) * e(r, q)$$
$$e(p, q+r) = e(p, q) * e(p, r)$$

Translation: you can pull additive component out of a pairing by multiplication

(2/24) Forgive me, dear lord, for what I am about to do to the math.



(3/24) If you've gotten this far, you know how ridiculously difficult the math has already been. I would call it pretty far beyond advanced, right?

I literally laughed out loud when I got to this part in @VitalikButerin's blog, how did you react?

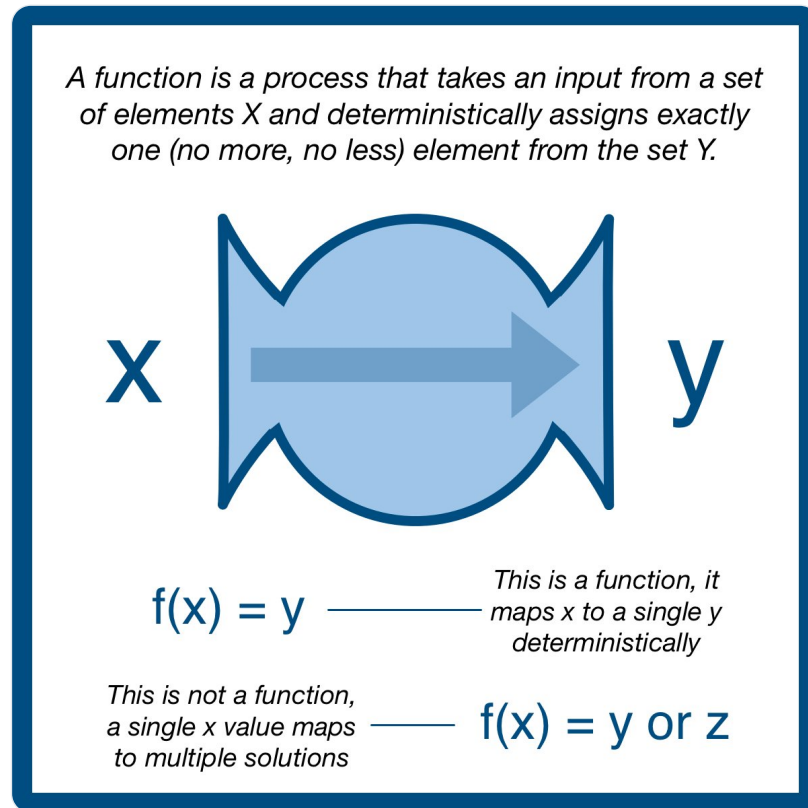
So how do we do this?

The math behind why pairing functions work is quite tricky and involves quite a bit of advanced algebra going even beyond what we've seen so far, but I'll provide an outline.

(4/24) Ok we'll start small: functions.

A function is a process that takes an input from a set of elements  $X$  and deterministically assigns exactly one (no more, no less) element from the set  $Y$ .

We write a function as  $f(x) = y$ ; the function  $f$ , when applied to  $x$ , produces  $y$ .



(5/24) Functions might be familiar, the term "set" might be new. A set is simply a collection of different things:

Set of all integers (infinite):  $\{\dots, 0, 1, 2, 3\dots\}$

Set of even integers (infinite):  $\{\dots, 2, 4, 6, 8\dots\}$

Set of Haym's favorite numbers (finite):  $\{4, 13\}$

(6/24) Mathematical sets and set theory is an incredibly important area of math; but we aren't really here to discuss sets.

For our purposes we just need to understand that functions can move from an input set to a different output set.

(7/24) If an output set/group is different from an input set/group, we gain a very specific property: the output of this kind of function cannot be used as an input.

Put another way, you can only execute these kinds of functions one time; you cannot execute them repeatedly.

(8/24) This property is incredibly important for cryptology; it breaks most efficient methods of cracking crypto schemes and forces an attacker to use the slow difficult method: guess and check.

So, let's build an elliptic curve pairing!

(9/24) An elliptic curve pairing is a function that takes in two points on an elliptic curve and outputs a point in a finite field (think finite field = a predetermined set of points).

Think of a pairing as the elliptic curve equivalent of multiplication.

*Elliptic curve pairings are the elliptic point equivalent of multiplication*

$$e(p, q) = t$$
$$\approx$$
$$p * q = t$$


*p, q are points on an elliptic curve, t is a point in a finite field*


(10/24) Not all elliptic curves have pairings, and not all pairings are secure. But once found, a single pairing can be shared by everyone.

So while us plebs are trying to get through tweets on intro-level topics, the chads like [@danboneh](#) are working on developing more pairings.

(11/24) Here's the plan: we are going to take two (different) elliptic curves, both of which are hiding a secret number  $S$  within their structured reference string (eg step 1 of a KZG commitment).

Once we have our two points, we will apply the pairing to get one final value.

**Haym**  
@SalomonCrypto · Follow



(1/23) KZG Commitments Part 1: Commit

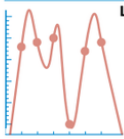
Our goal: 1) prove we are committed to specific data and 2) allow others to verify specific points within that dataset.

Before we get started, we need to prepare the data and elliptic curve.... Then, it's time for some magic!

### KZG Polynomial Commitments

Red: Secret   Green: Public   Blue: Elliptic Curve (Public)   Pink: Data (Varies)

#### Step 0: Preperation


shared between all commitments		specific to each commitment	
Trusted Setup		Data	
Secret Number $S$	Elliptic Curve: $y^2=x^3+ax+b$	Plaintext Data $STUART$	UTF-8 Encoding: 83, 84, 85, 65, 84
$f(S^0) = [S^0] = S^0G$ $f(S^1) = [S^1] = S^1G$ $f(S^2) = [S^2] = S^2G$ $f(S^3) = [S^3] = S^3G$ $\vdots$ $f(S^n) = [S^n] = S^nG$			Lagrange Polynomial: $f(x) = a_0x^0 + a_1x^1 + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$
Public Structured Reference String (SRS)			




#### Step 1: Commit

Commitment:  $[f(S)]$  — single value that serves as the polynomial commitment

$$[f(S)] = [a_0S^0 + a_1S^1 + a_2S^2 + a_3S^3 + a_4S^4 + a_5S^5]$$
$$[f(S)] = a_0[S^0] + a_1[S^1] + a_2[S^2] + a_3[S^3] + a_4[S^4] + a_5[S^5]$$

1:06 AM · Oct 19, 2022

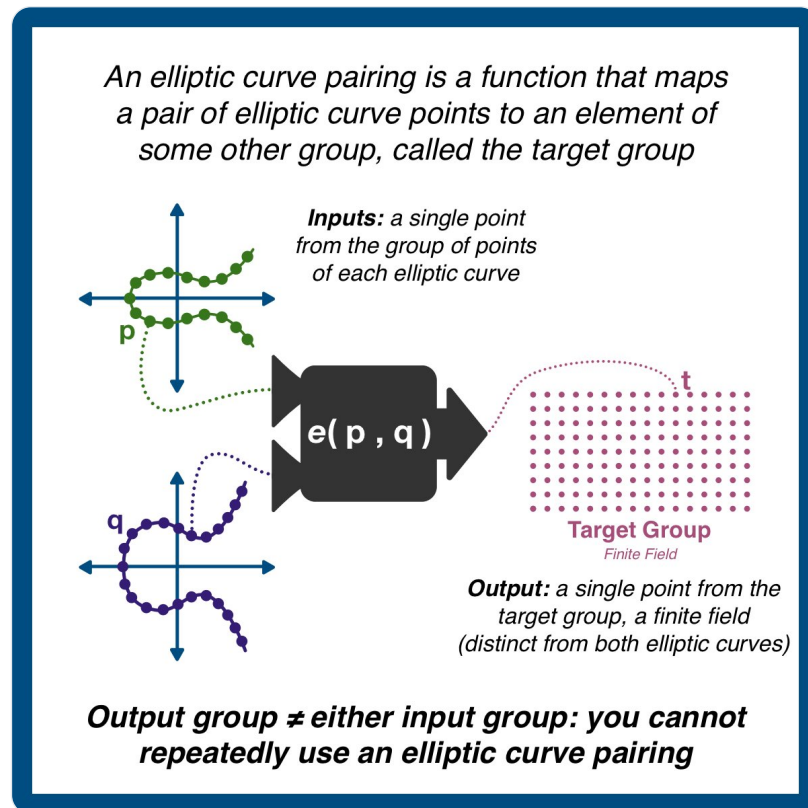
 Read the full conversation on Twitter

 183    Reply    Copy link

Read 5 replies

(12/24) While advanced math is difficult to visualize, the image below should help clarify.

The pairing (denoted as  $e(p,q)$ ) takes a point from each elliptic curve and outputs  $t$ .  $t$  is NOT a point on either curve, and therefore the pairing cannot be used more than once.



(13/24) If all you care about is how we are going to use pairings, you can stop here. We are simply going to use them like multiplication: combine two points to make a third.

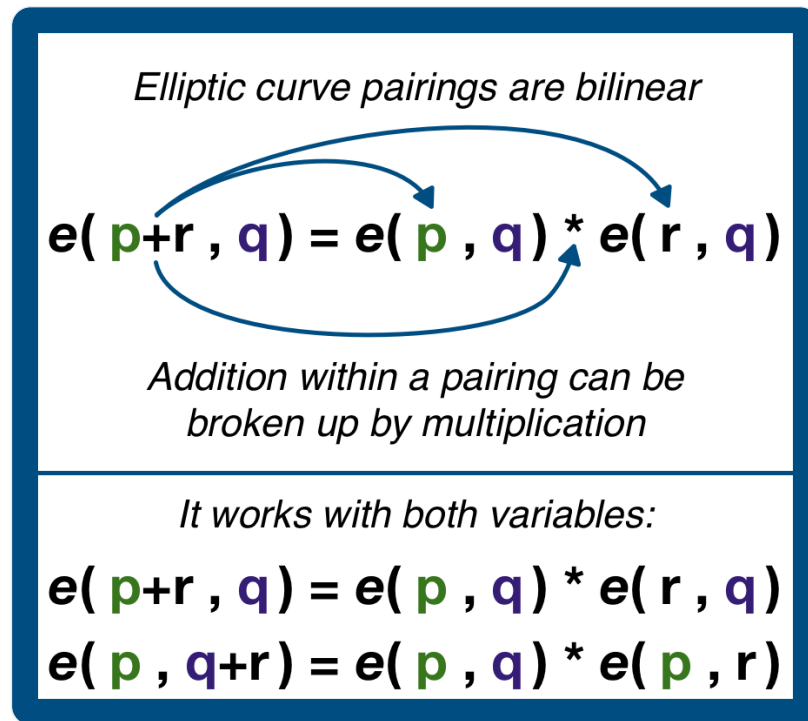
We will use the pairing two times with different values and compare the results, looking for equality.

(14/24) The rest of this thread will BRIEFLY touch on the HIGHEST levels of pairings.

Honestly, this probably won't mean anything to you unless you already know relatively advanced algebra/number theory. But, it's worth including just for completeness.

(15/24) First, the most important property of pairings is that pairings are bilinear.

Bilinear is one of those arcane, hard-to-unpack terms. Suffice to say that it is a specific algebraic property that allows us to expand/contract an expression.



(16/24) Pairings do not break 2 of the 3 properties of Diffie-Hellman:

Still incredibly difficult:

- Discrete log problem (given  $g^x$ , solve for  $x$ )
- Computational Diffie-Hellman (given  $g^x$  and  $g^y$ , find  $g^{xy}$ )

However, decision Diffie-Hellman becomes very easy.

(17/24) Decision Diffie-Hellman is the property that says given  $g^x$ ,  $g^y$ , and  $g^z$ , it should be hard to see if  $xy = z$ .

The bilinear property of elliptic pairing break this assumptions, it is trivial to test if  $e(g^x, g^y) = e(g^z)$ .

(18/24) Fortunately, pairings are cyclic, meaning that as the input values grow, the output values eventually wrap back around.

And, just like all the modular arithmetic problems we work in, undoing the operation is realistically impossible.

(19/24) And so, if we modify the decision Diffie-Hellman problem to a bilinear Decision Diffie-Hellman problem, pairings hold all the same properties and assumptions required to implement Diffie-Hellman and most other public cryptography schemes.

(20/24) There is one more property of pairings that is particularly useful: the decision linear assumption (DLIN).

DLIN is incredibly abstract (it has to do with being able to distinguish the linear combination of the exponents of a generator vs a random exponent).

(21/24) (Pause)

The next tweet is going to be the biggest sin I've committed against the study of mathematics in this entire thread.

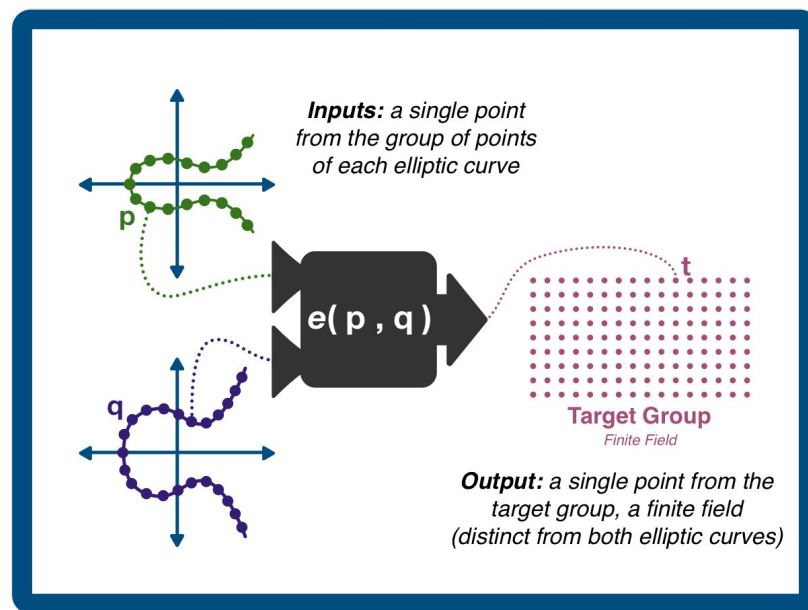
Forgive me, dear Newton...

(Resume)

(22/24) DLIN is a property about comparing the results of applying two matrices to the exponent (take you number  $n$  and you matrix  $A$  and create a new matrix where each matrix element is  $n^{[\text{matrix element}]}$ ).

If the rank of matrix  $A$  is less than matrix  $B$ ,  $B$  will appear random vs  $A$ .

(23/24) Rough idea: the target group is has more elements than either of the input groups (the target group has higher order or rank), therefore feeding inputs produces outputs that appear to be random (even if you feed them in sequential order).



(24/24) Bottom line: elliptic curve pairings are HARD. This is the bleeding edge of math research... welcome to the workshop, where they make magic out of algebra!


Here is the main take away:

A pairing an irreversible, one time elliptic curve point multiplication.




Like what you read? Help me spread the word by retweeting the thread (linked below).

Follow me for more explainers and as much alpha as I can possibly serve.



**Haym**  
@SalomonCrypto · [Follow](#)



(1/24) Degen's Handbook: A Practical Guide to Elliptic Curve Pairings

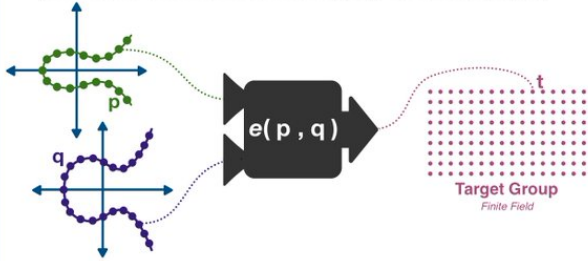
Magic is real, elliptic curve pairings are proof. Through incredibly advanced math, pairings allow us to multiply two polynomials... through elliptic curves!

Here's all you need to know.

**Elliptic curve pairings are the elliptic point equivalent of multiplication**

$$e(p, q) = t \approx p * q = t$$

An elliptic curve pairing is a function that takes a pair of elliptic curve points and returns an element of some other group, called the target group





Think of a pairing as a black box that takes elliptic points. Pairings cannot be used consecutively; the target group points don't match the input points




Elliptic curve pairings are bilinear, holding to the following property:

$$e(p+r, q) = e(p, q) * e(r, q)$$
$$e(p, q+r) = e(p, q) * e(p, r)$$

Translation: you can pull additive component out of a pairing by multiplication

6:50 PM · Oct 20, 2022 

 [Read the full conversation on Twitter](#)

  [Reply](#)  [Copy link](#)

[Read 1 reply](#)

...