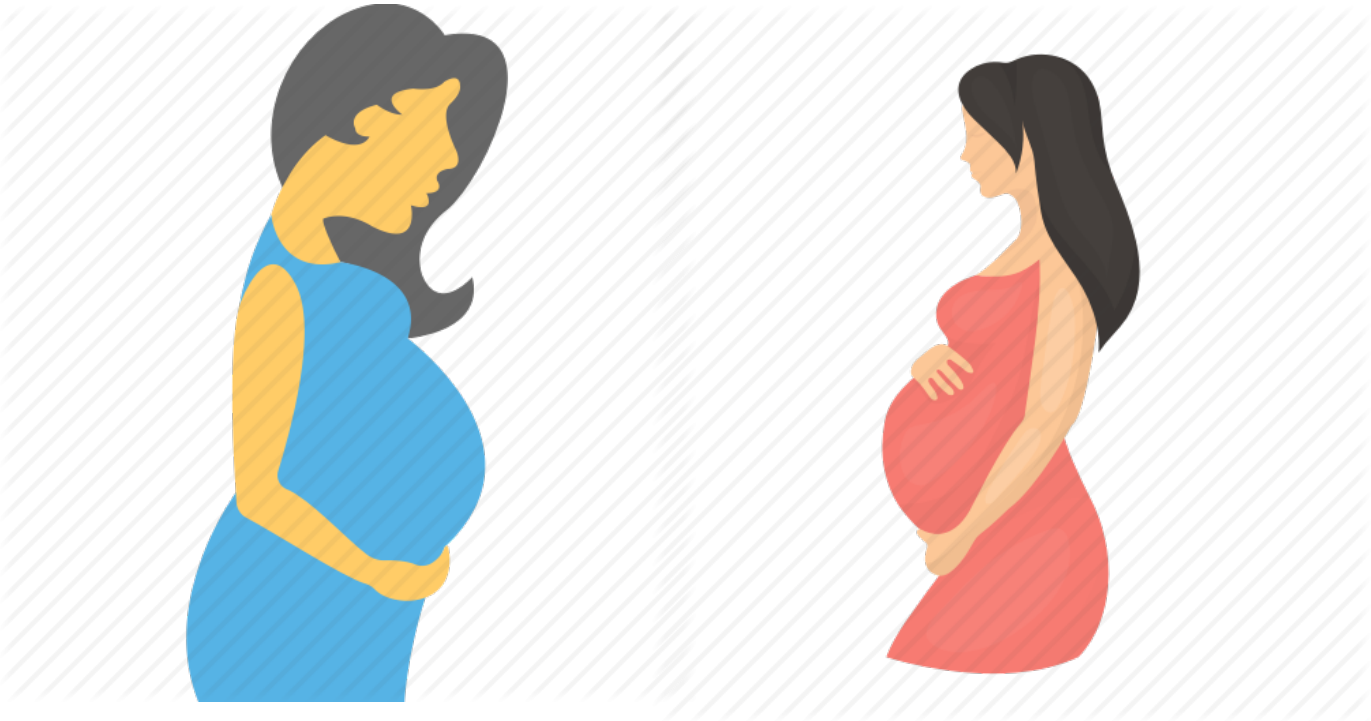
On Verifiable Delay Functions (VDF):

How to Slow Burning Down the Planet (Verifiably)

 @asanso

Joint work with De Feo, Masson, Petit

Can two women have a baby in 4.5 months?



Agenda

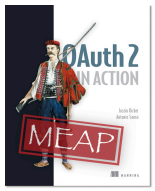
- Definition
- Applications
- Constructions
- Conclusions

Who is this guy, BTW?

Security Researcher @Adobe Research Switzerland

Google Security Hall of Fame, Facebook Security Whitehat, GitHub Security Bug Bounty, Microsoft Honor Roll, etc

Found vulnerabilities in OpenSSL ,Google Chrome, Safari



Co-Author of “OAuth 2 in Action”

RUHR UNIVERSITÄT BOCHUM RUB Phd Student Ruhr Universität

$2_3\ 5_7\ 11$ Obsessed by prime numbers

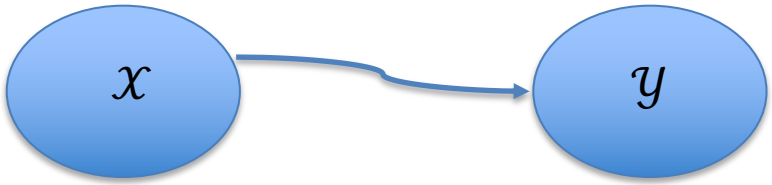
What is a VDF?

A function that:

1. Takes **T steps** to evaluate even with **unbounded parallelism**
2. The output can be **verified efficiently**

What is a VDF?

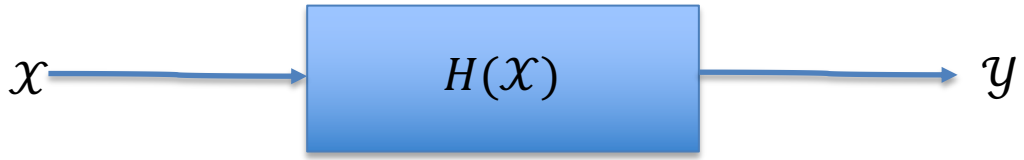
- **F**unction



- **D**elay

- **V**erifiable

Cryptographic Hash functions



- **Deterministic**
- **Hard** to guess
- **Infeasible** to find two different messages with the **same hash value**
- **Infeasible** to generate a message that yields a **given hash value**

VDF minus any property is “easy”

- **Not V**erifiable :

$$s \to H(s) \to H(H(s)) \to \cdots \to H^{(T)}(s) = a.$$

- **No D**elay : Easy (many example in cryptography e.g. Discrete Log)

- **Not F**unction : Proof of sequential work

VDF

- Setup
- Evaluation
- Verification

VDF Application #1

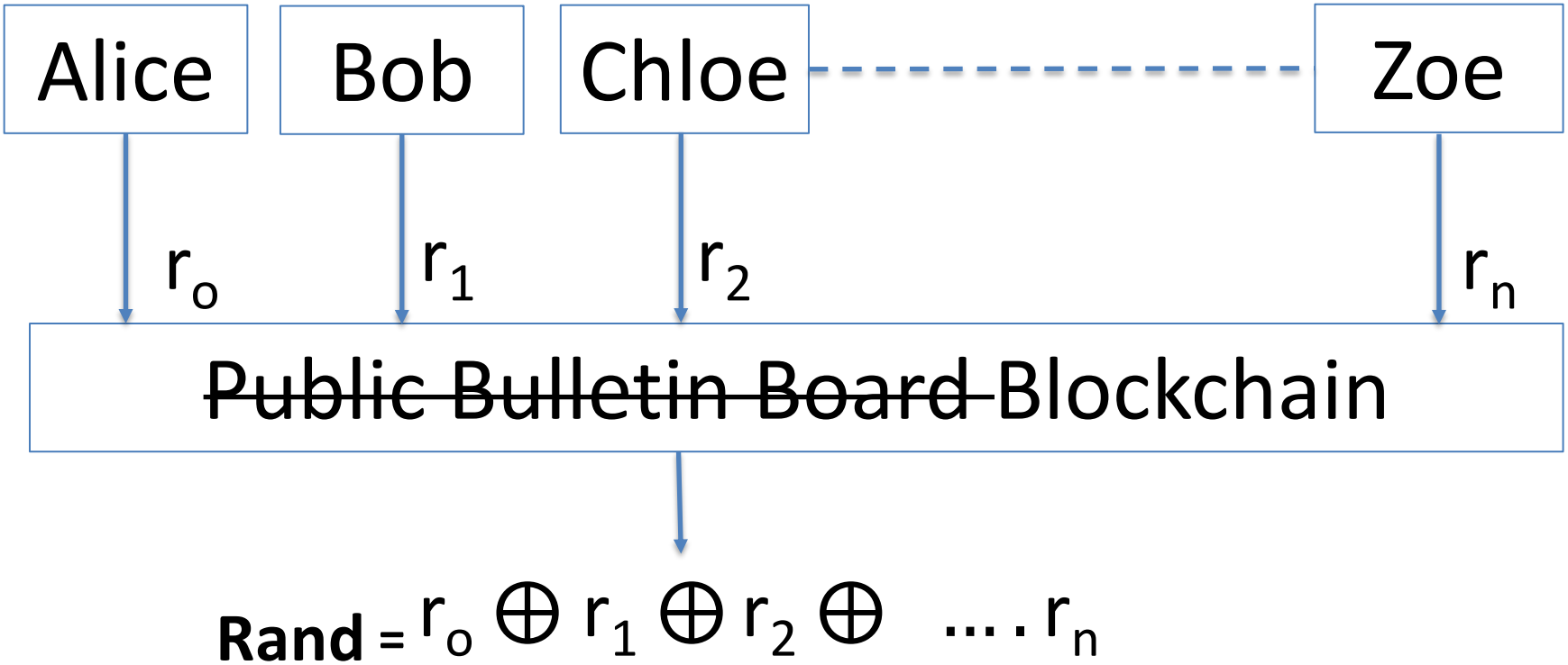
Generate **verifiable** randomness

Only 3 balls drawn!

Five Numbers Shown!!

VDF Application #1

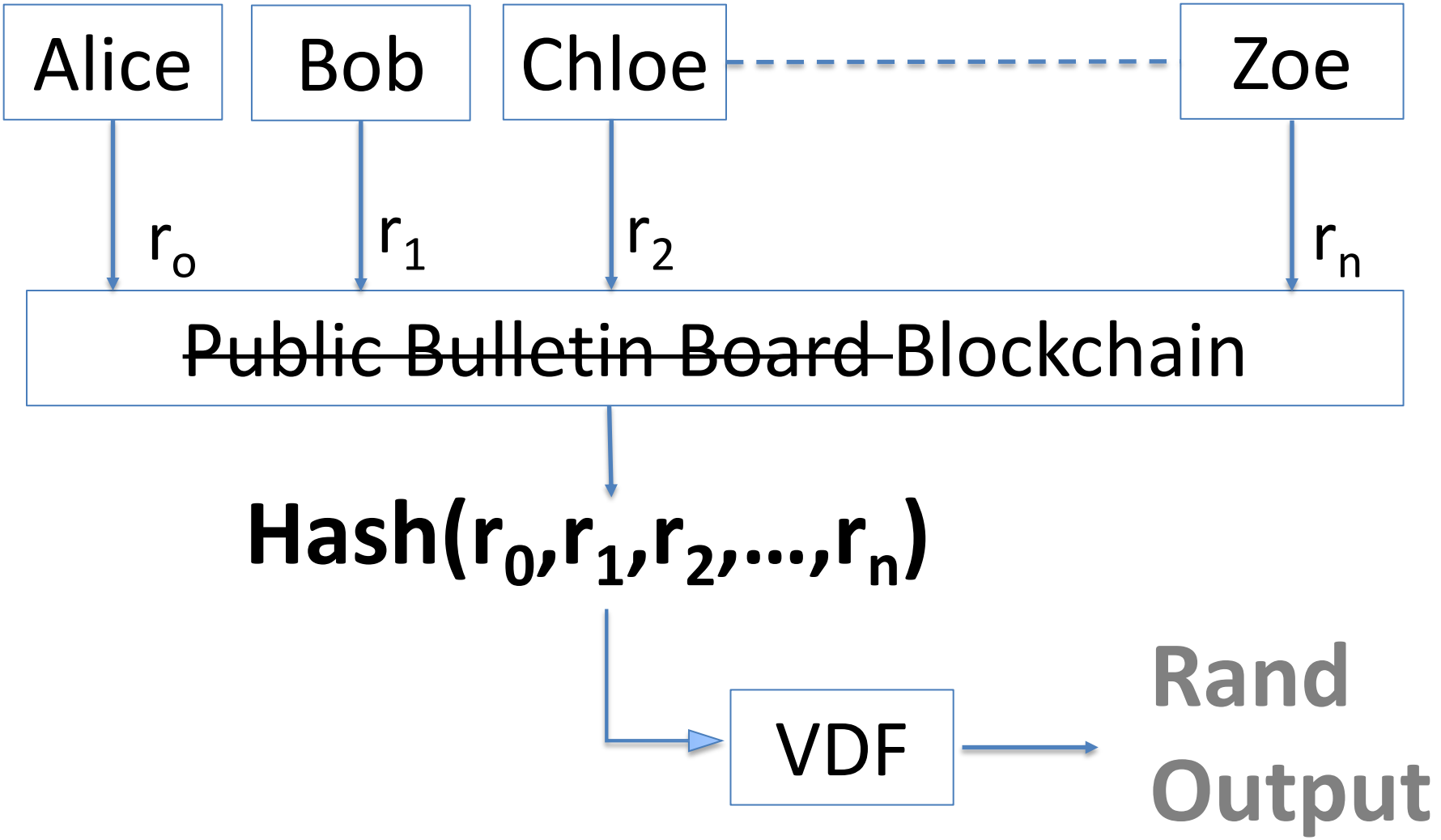
Distribute generation (broken)



Problem: Zoe has controls of the output

VDF Application #1

Distribute generation



VDF Application #2

Blockchains

BITCOIN WILL BURN THE PLANET DOWN. THE QUESTION: HOW FAST?

VDF Application #2

Blockchains

Ethereum Betting $15 Million on VDF tech for Serenity

by Supriya Saxena (Senior Correspondent) February 7, 2019

SHARE 0

Chia VDF competition and implementation

In an attempt to create a secure, open and decentralized consensus algorithm, Chia is hosting a 3 month long competition, with a total of around $100,000 in prizes, for the implementation of fast and secure verifiable delay functions using class groups.

VDF Application #2

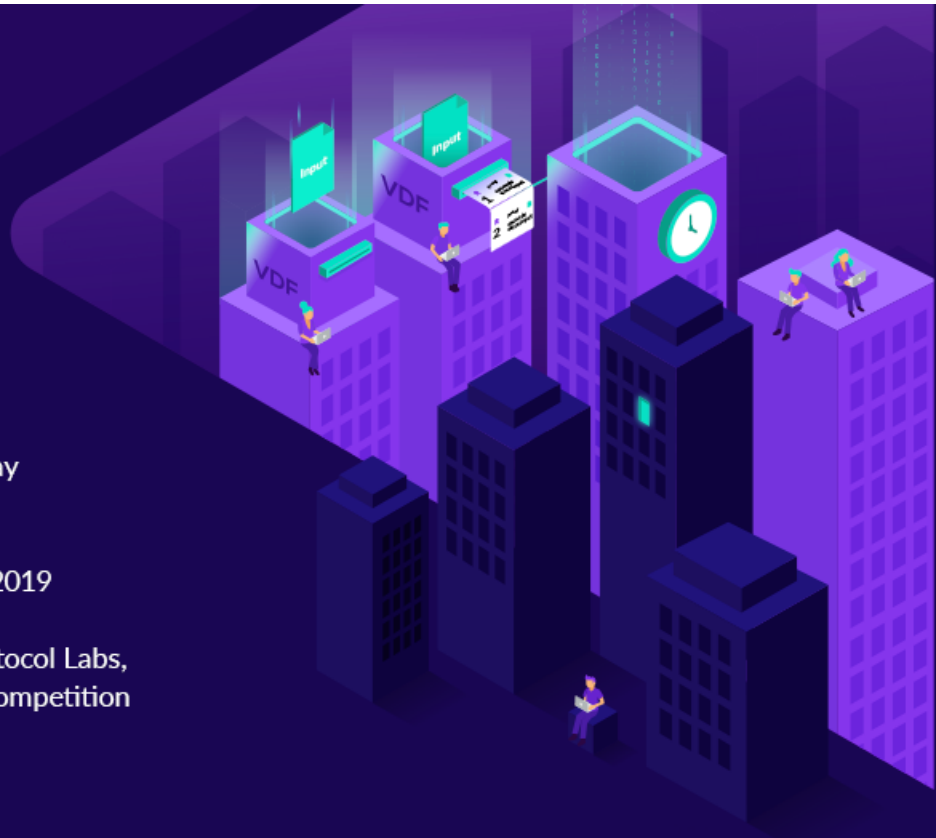
Blockchains

VDF | FPGA Design Competition

Are you up for the challenge?

$100,000 Prize to Forever Change Blockchain

- The problem: Given 1024-bit input x, compute the verifiable delay function h=x^(2^t) mod N as fast as possible.
- Timeline: The overall competition will run from Aug 1 - Dec 30, 2019
- Prize: The Ethereum Foundation, the Interchain Foundation, Protocol Labs, Supranational, Synopsys, and Xilinx are sponsoring a $100,000 competition with support from AWS.



Read more
https://www.vdfalliance.org/contest

ethereum foundation
SUPRA NATIONAL
INTERCHAIN FOUNDATION
Protocol Labs

VDF History

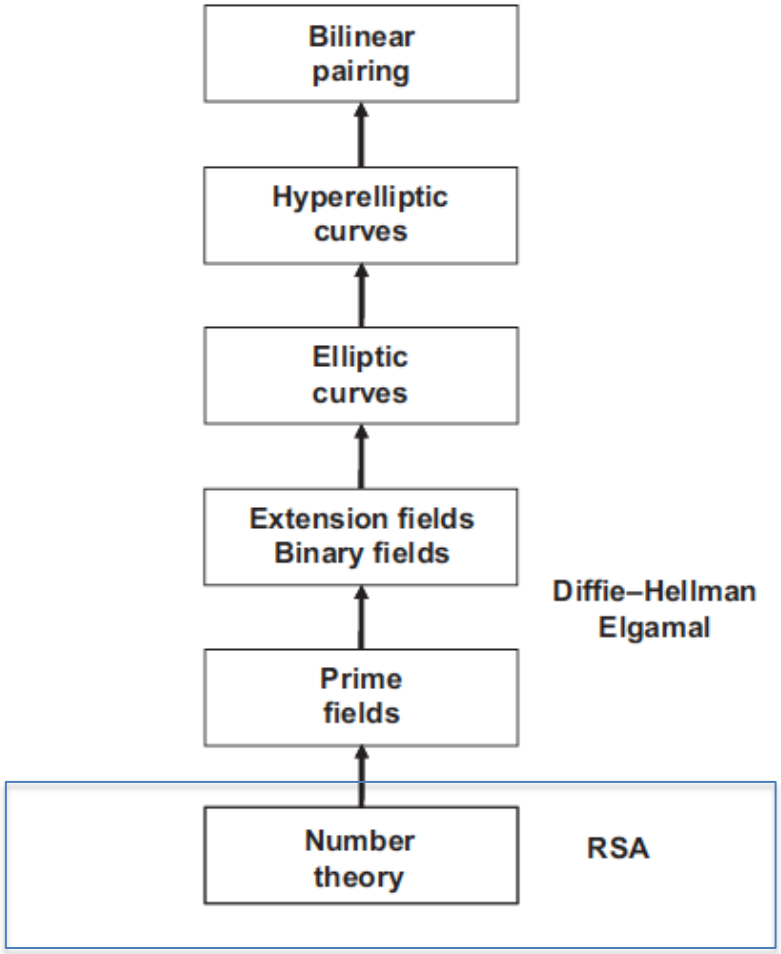
https://vdfresearch.org/

- 2018 (12 June) : Seminal paper by Boneh, Bonneau, Bünz, Fisch (BBBF), no actual VDF implementation
- 2018 (20 June) : Wesolowski's VDF
- 2018 (22 June) : Pietrzak's VDF
- 2019 (20 February) : Isogenies VDF by De Feo, Masson, Petit, Sanso (FMPS)


VDF
VDF EVERYWHERE


CAUTION
MATH
AHEAD

VDF #1 and #2 Wesolowski & Pietrzak



RSA Refresher

N = p * q (p and q big prime numbers)

e public exponent (e.g. 65537)

Encryption

$secret^{e} \pmod{N}$

Time Lock puzzle (RSW '96)

N = p * q (p and q big prime numbers) and **keep p and q secret** (group of unkown order)

Evaluate $s^{2^T} \pmod N$

With 2^T being huge

Caveat whoever knows the factorization of N can cheat.

Order= $\varphi(N) = (p-1)(q-1)$

How? ➔ $\mu = 2^T \pmod{\varphi(N)}$

Compute s^{μ} *instead*

MIT LCS35 Time Capsule Crypto-Puzzle

- Designed by **Ron Rives** in 1999: "*We estimate that the puzzle will require 35 years of continuous computation to solve*"
- Solved by **Bernard Fabrot** in 2019 (3.5 years of computation)
- Almost concurrently solved by a team at **Supranational** (led by **Simon Peffers**) using a novel squaring algorithm (ran for 2 months!!!), designed by **Erdinç Öztürk** from Sabanci University

Wesolowski
&
Pietrzak

Time Lock Puzzle

+

Fast Verification

(without revealing the group's order)

Wesolowski's VDF (Interactive version)

Given *(g,h)* Alice wants to prove to Bob that $h = g^{2^T}$

Alice		**Bob**
	l ←	Choose a random prime *l*
Find *q* and *r* s.t. $2^T = ql+r$	$\pi = g^q$ →	
		Compute $r = 2^T \pmod l$
		Accept if $\pi^l g^r = h$

Why? ➔ $\pi^l g^r = g^{ql} g^r = g^{2^T} = h$

Wesolowski's VDF (Non interactive version)

Apply **Fiat-Shamir** heuristic

$l = next\_prime(hash(g,h,T))$

Pietrzak's vs. Wesolowski's VDF

Pietrzak	**Wesolowski**
Faster	Faster
Proof	Proof
Computation	Verification
	Shorter proof

Groups of Unknown Order

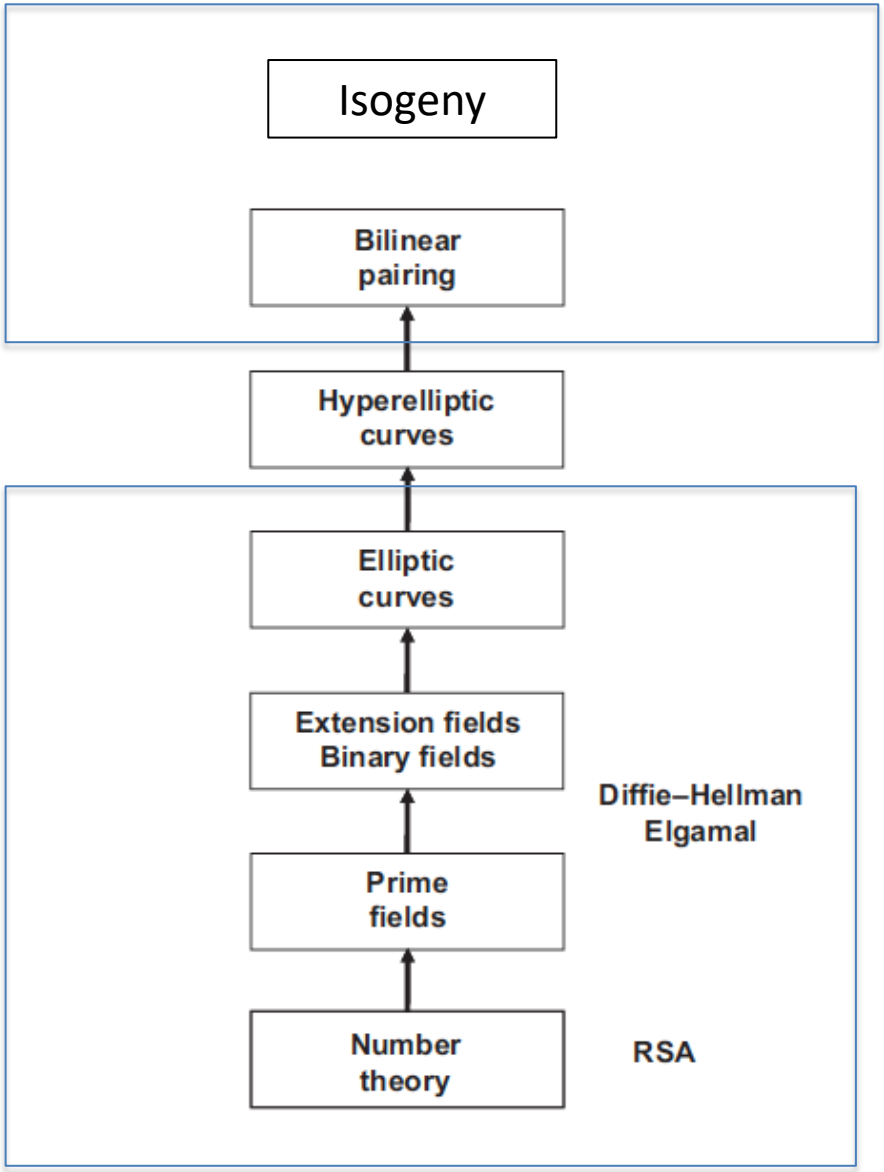
- RSA group → Needs **trusted setup**!

- RSA UFO 

(*Unknown Factorization Objects*) → **Expensive** (N ~ 30k bit)!

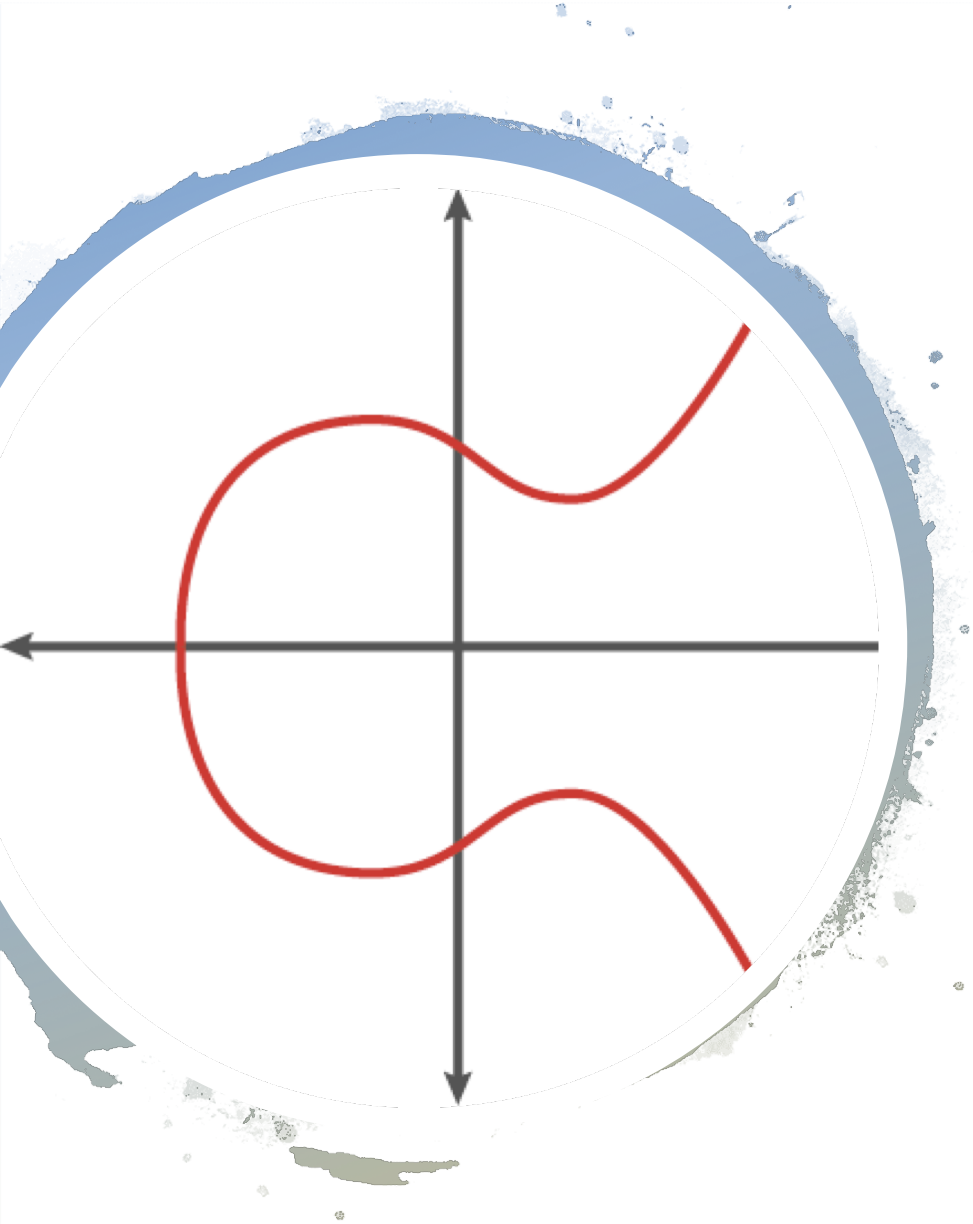
- Class groups of imaginary quadratic field → **No trusted setup** a **bit slower** than plain RSA

Isogeny VDF (De Feo, Masson, Petit, Sanso)



History of Elliptic Curve (Cryptography)

- **Diophantus** (Arithmetica ~3rd century AD)
- **Henri Poincaré** (1901)
- **André Weil** (1929)
- **Hendrik Lenstra** (1984)
- **Koblitz** and independently **Miller** (1985)

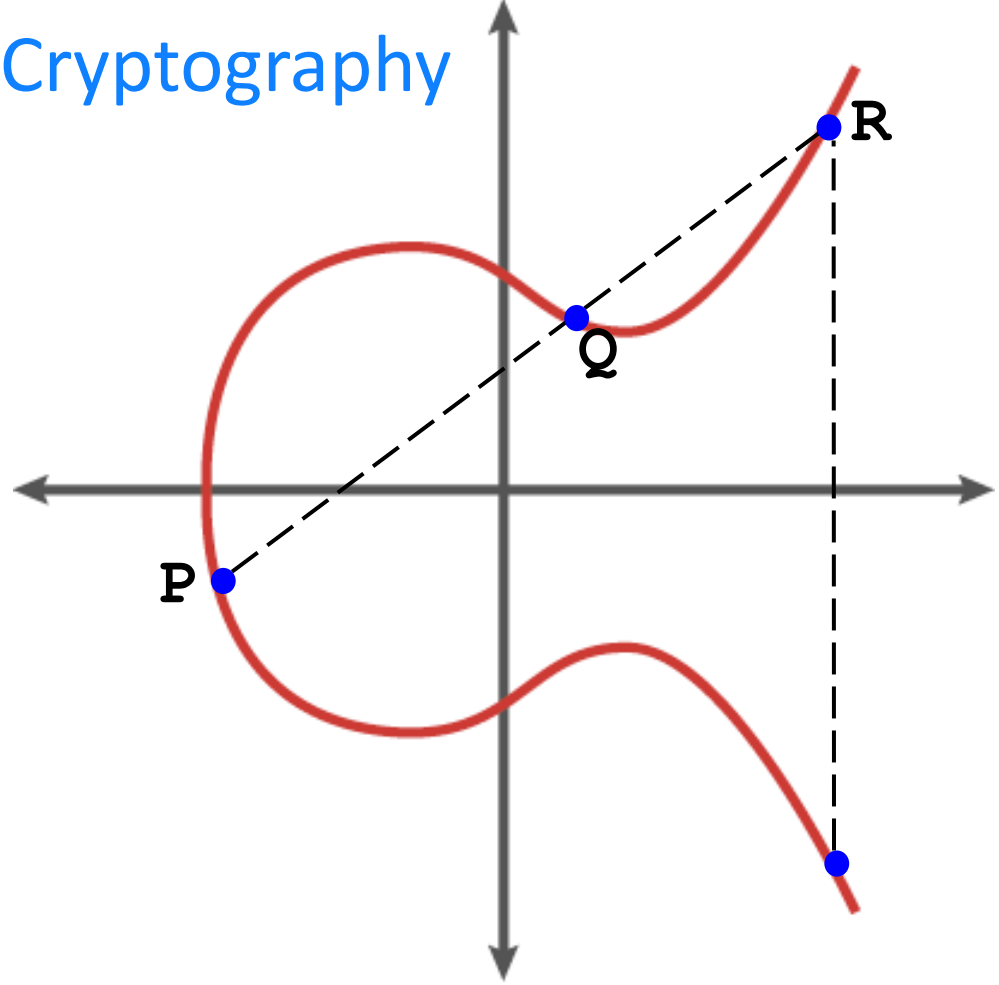


What is an Elliptic Curve

An elliptic curve is the set of solutions defined by an equation of the form

$$y^2 = x^3 + ax + b$$

Elliptic Curve Cryptography



Website Identity

Website: www.adobe.com

Owner: This website does not supply ownership information.

Verified by: DigiCert Inc

Expires on: February 5, 2020

Privacy & History

Have I visited this website prior to today? Yes, 50 times

Is this website storing information on my computer? Yes, cookies and 1.0 MB of site data

Have I saved any passwords for this website? No

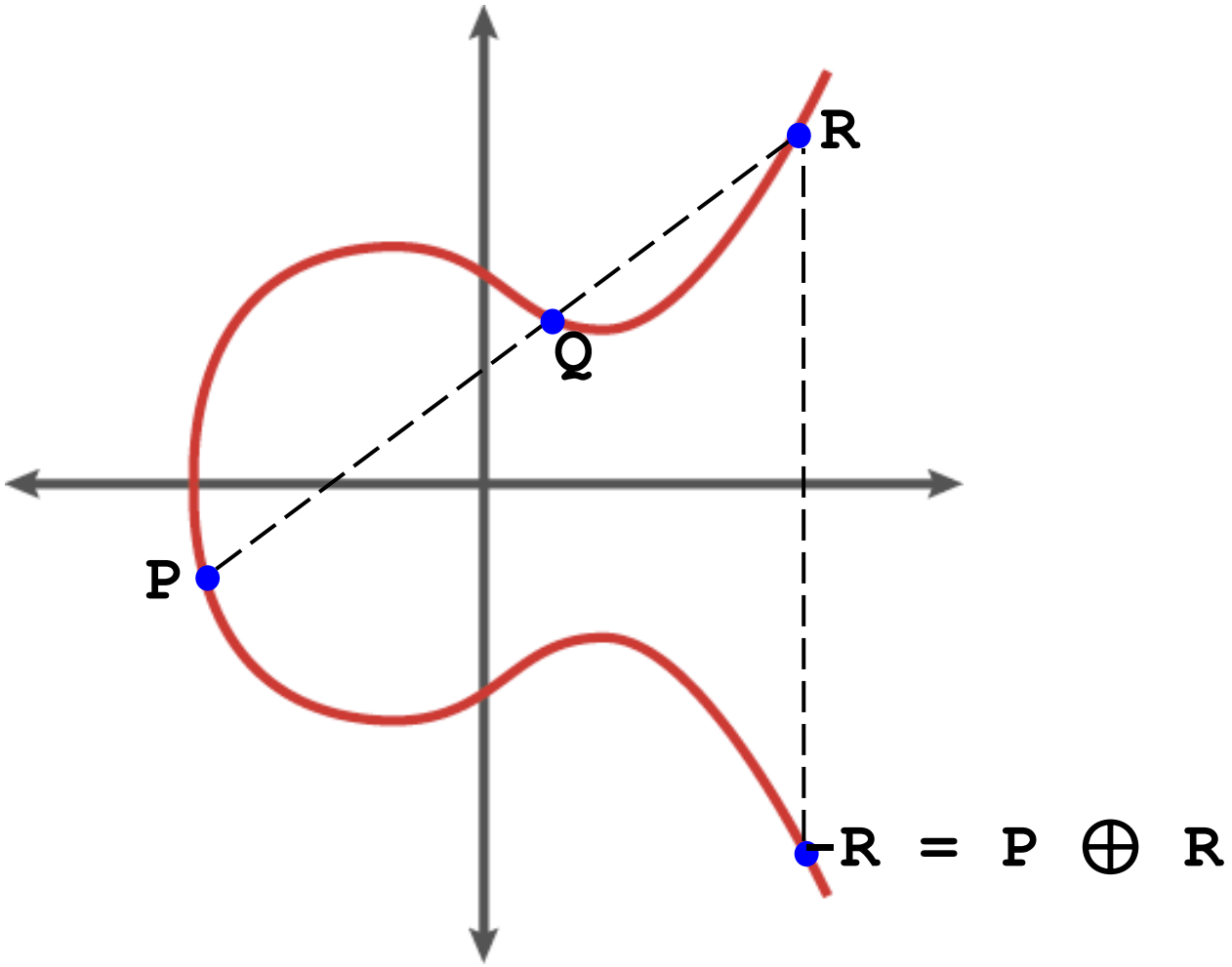
Technical Details

Connection Encrypted (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, 256 bit keys, TLS 1.2)

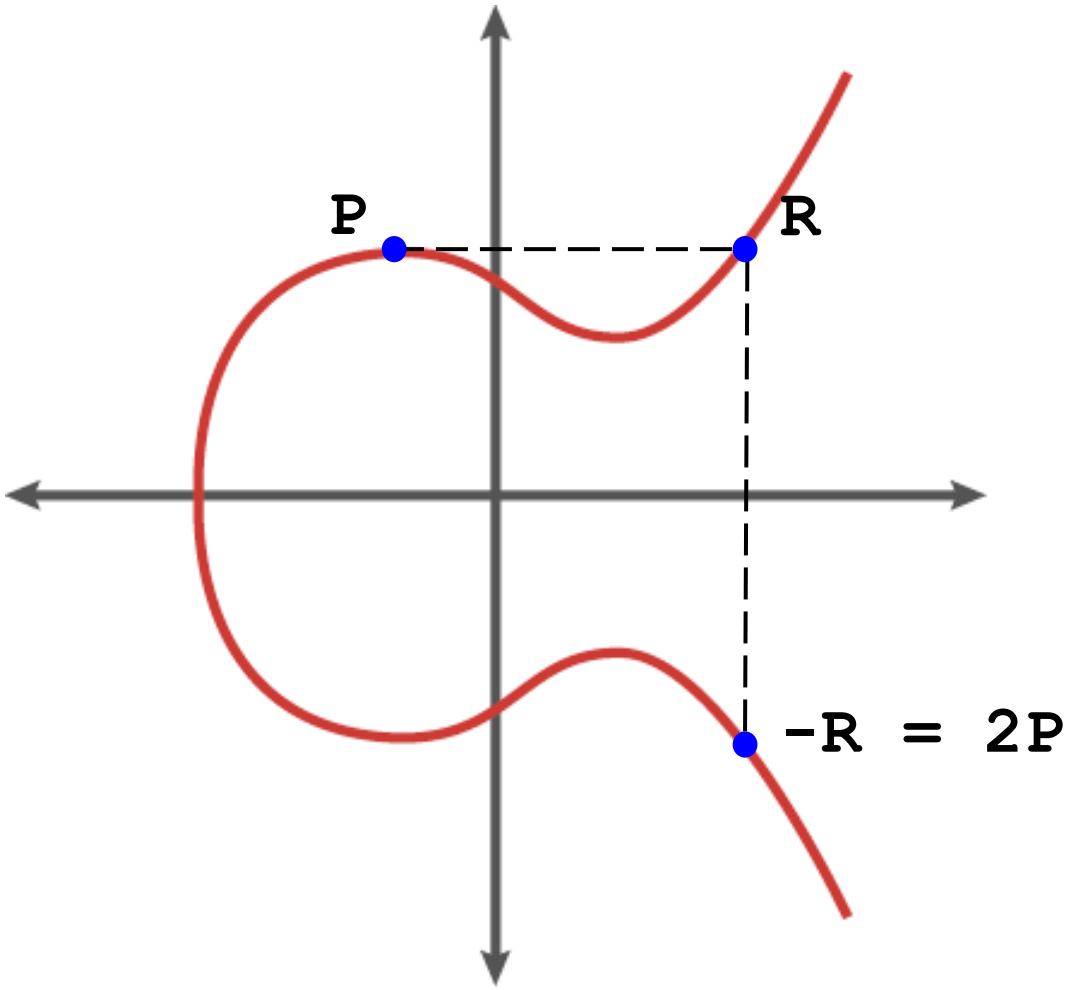
The page you are viewing was encrypted before being transmitted over the Internet.

Encryption makes it difficult for unauthorized people to view information traveling between computers. It is therefore unlikely that anyone read this page as it traveled across the network.

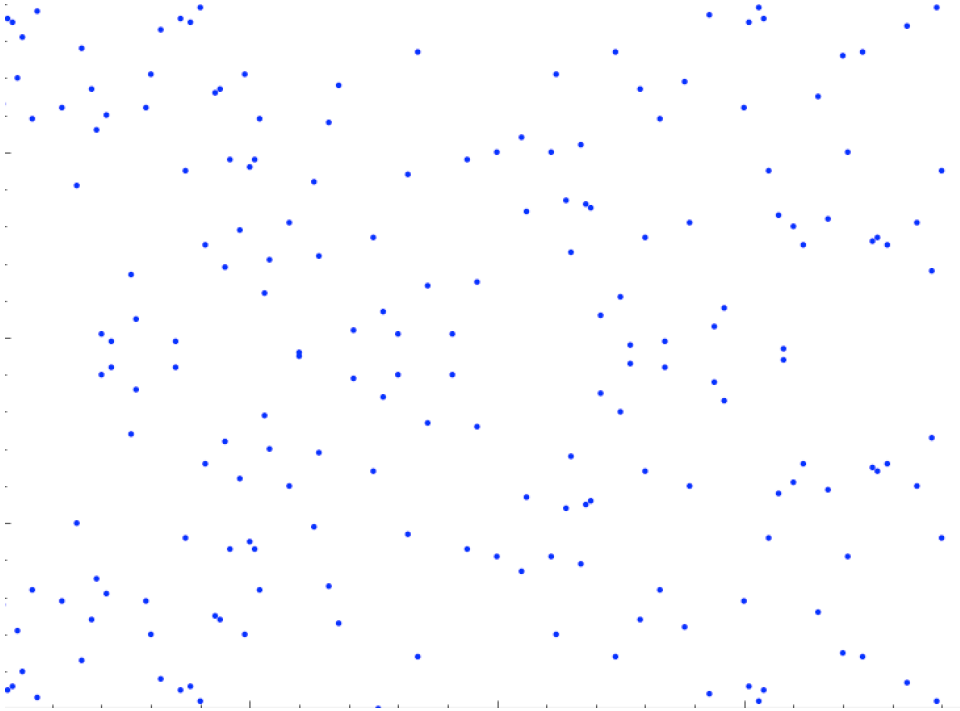
Elliptic Curve Addition



Elliptic Curve Point Multiplication



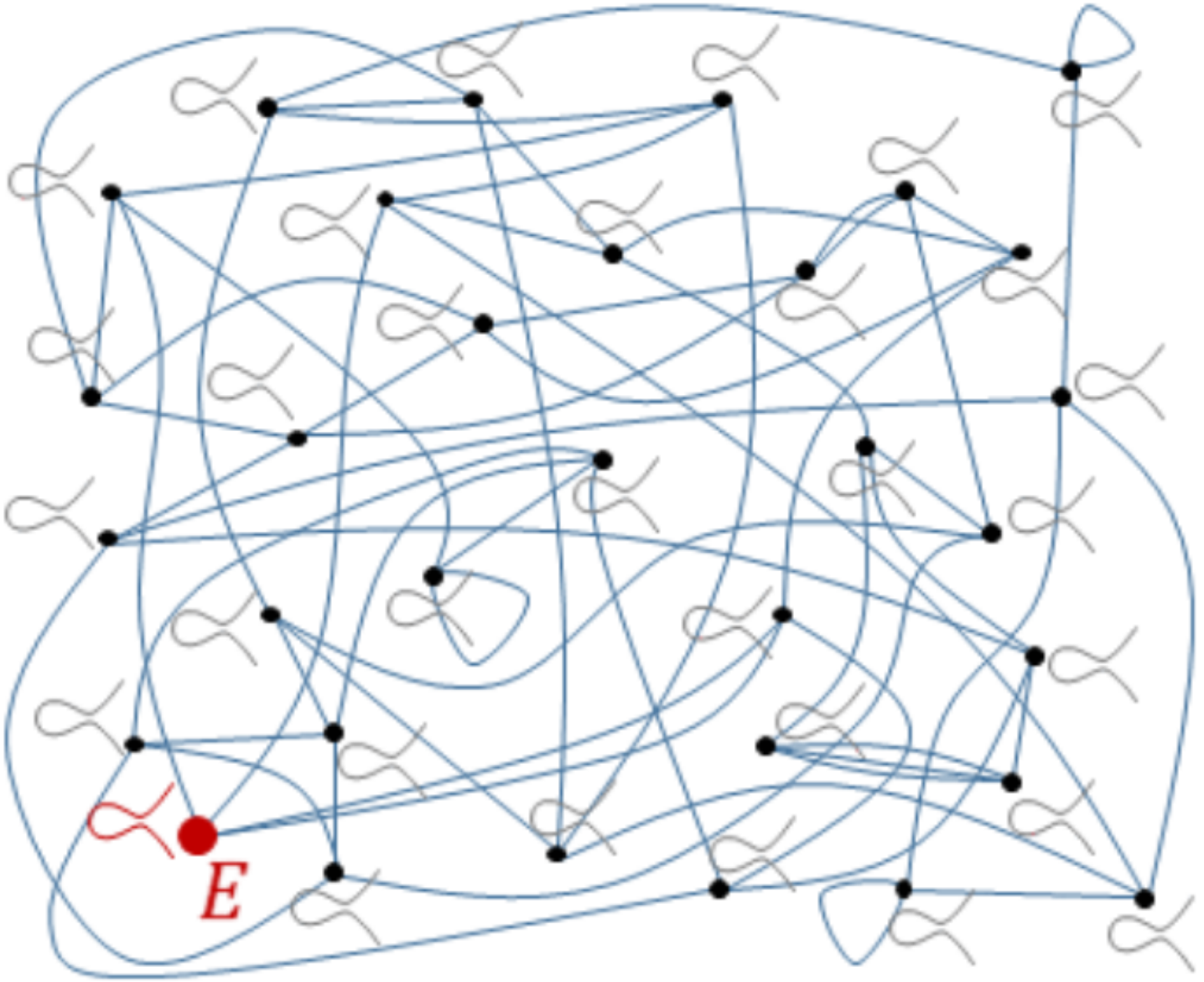
Elliptic Curve over Finite Fields



$$y^2 = x^3 + 4x + 20$$

over Finite Field of size 191

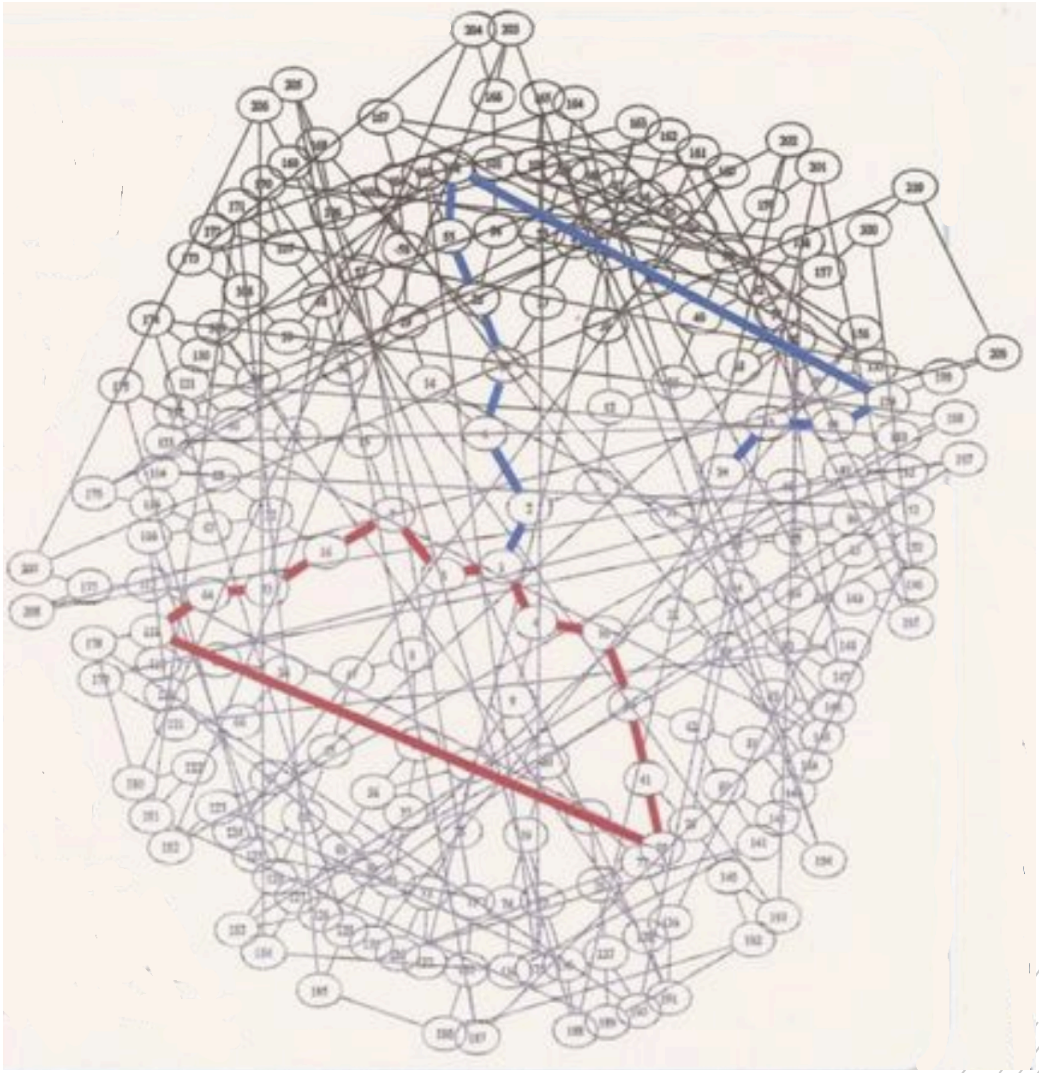
Isogeny Based Cryptograhy



History of isogeny-based cryptography

- 1996: Couveignes introduces isogeny in cryptography (**paper rejected Eurocrypt**)
- 2006: Rostovtsev & Stolbunov independently rediscover Couveignes ideas
- 2007: Charles, Goren & Lauter propose supersingular for a “provably secure” hash function
- 2011: Jao, **De Feo** introduce SIDH, an efficient post-quantum key exchange (**SIDH**)
- **2012: …**

Isogeny VDF



© science.sciencemag.org

VDFs from isogenies and pairings[4]

$$\begin{array}{ccc} X_1 \times Y_2 & \xrightarrow{\phi \times 1} & Y_1 \times Y_2 \\ \Big\downarrow{\scriptstyle 1 \times \hat{\phi}} & & \Big\downarrow{\scriptstyle e'_N} \\ X_1 \times X_2 & \xrightarrow[e_N]{} & \mathbb{F}_{p^k} \end{array}$$

Setup:
- Supersingular curve $E/\mathbb{F}_p$ with (Weil) pairing e_N;
- Public isogeny $\phi : E \to E'$ of degree 2^T;
- The dual isogeny $\hat{\phi} : E' \to E$;
- A generator $\langle P \rangle = X_1 \subset E[N]$, compute $\phi(P)$.

Evaluate: On input a random $Q \in Y_2 \subset E'[N]$, compute $\hat{\phi}(Q)$.

Verify: Check that $e_N(P, \hat{\phi}(Q)) = e'_N(\phi(P), Q)$.

[4] De Feo, Masson, Petit, and Sanso 2019.

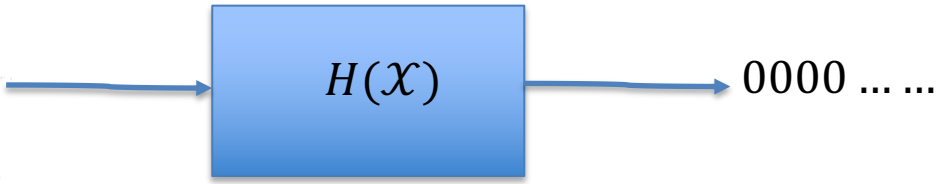
VDF Comparison

VDF	Sequential Eval	Parallel Eval	Verify	Setup	Proof size
Modular square root	T	$T^{2/3}$	$T^{2/3}$	T	—
Univariate permutation polynomials[6]	T^2	$> T - o(T)$	$\log(T)$	$\log(T)$	—
Wesolowski's VDF	$(1 + \frac{2}{\log(T)})T$	$(1 + \frac{2}{s \log(T)})T$	λ^4	λ^3	λ^3
Pietrzak's VDF	$(1 + \frac{2}{\sqrt{T}})T$	$(1 + \frac{2}{s\sqrt{T}})T$	$\log(T)$	λ^3	$\log(T)$
This work	T	T	λ^4	$T\lambda^3$	—
This work (optimized)	T	T	λ^4	$T\log(\lambda)$	—

Table 1. VDF comparison—Asymptotic VDF comparison: T represents the delay factor, λ the security parameter, s the number of processors. For simplicity, we assume that T is super-polynomial in λ. All times are to be understood up to a (global across a line) constant factor.

VDF Application #2
Blockchains

Proof of Work
(simplified)



Find x s.t.

$$H(x) = 0000 \ldots \ldots .$$

+ Random (VDF)

Ethereum 2.0

Proof of Stake +

Wesolowski's VDF

RSA Group

+ Multi Party Computation (MPC)

Chia

Proof of Space

+

Wesolowski's VDF

Class groups of imaginary quadratic field

Polkadot

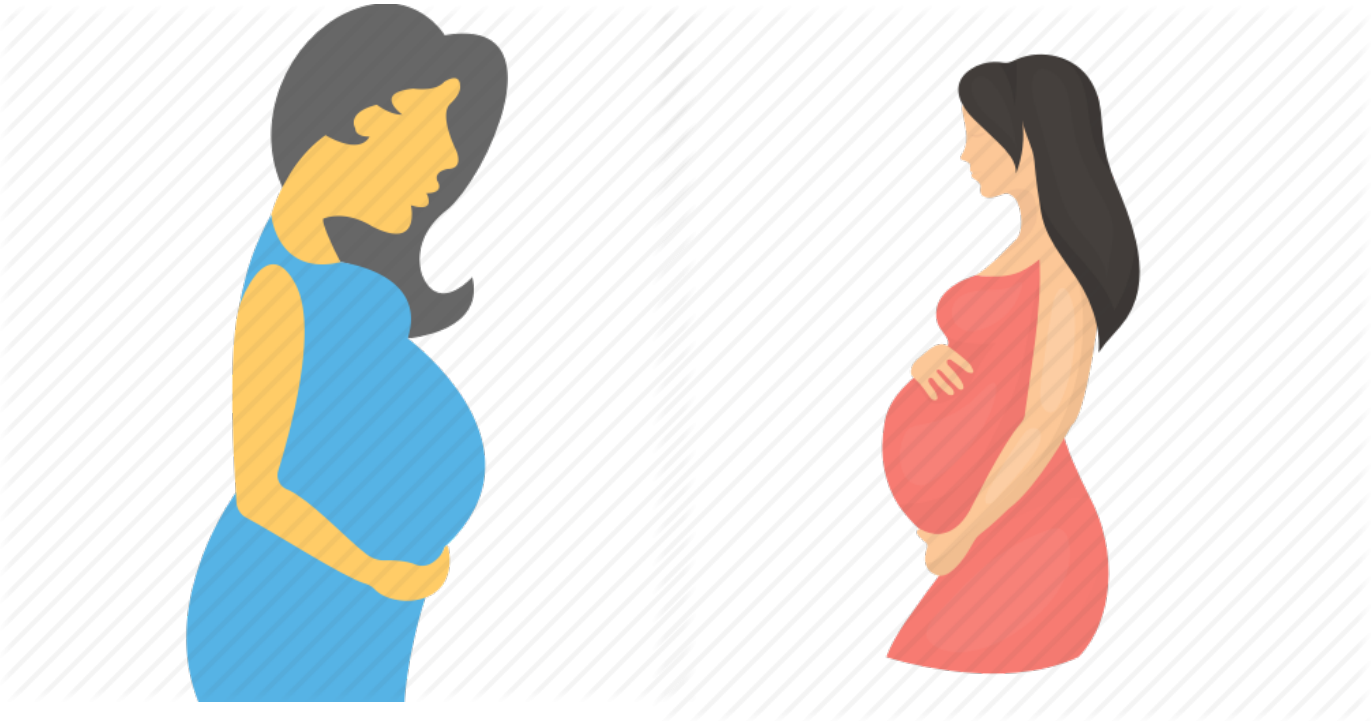
Isogenies

VDF ?

Other VDF Applications

- Protection again DOS
- CAPTCHA protection
- Workflow steps

Can two women have a baby in 4.5 months?



Questions?