Gems of TCS

Public Key Cryptography II

Sasha Golovnev

November 13, 2023

RSA

Modular Arithmetic

Easy Problems

- Addition, Subtraction, Multiplication
- GCD
- Modular Inverse
- Modular Exponentiation
- Primality Test

Modular Arithmetic

Easy Problems

- Addition, Subtraction, Multiplication
- GCD
- Modular Inverse
- Modular Exponentiation
- Primality Test

Hard Problems

- Factorization
- eth root: $x^{1/e}$

Euler's Theorem

Euler's Function

$$\forall N \in \mathbb{N},$$
$$\phi(N) = \#\text{ of invertible els in } Z_N$$
$$= |\{x : GCD(x, N) = 1\}| .$$

Euler's Theorem

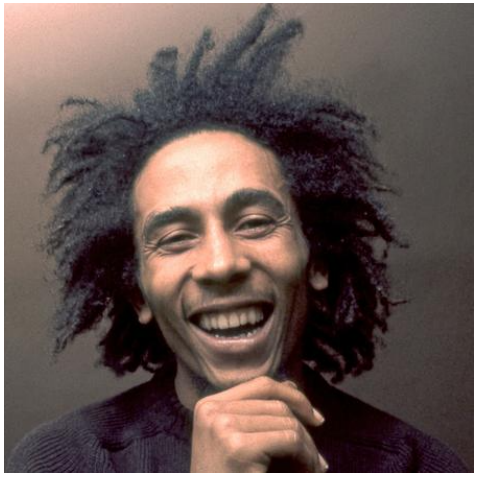
Euler's Function

$$\forall N \in \mathbb{N},$$
$$\phi(N) = \#\text{ of invertible els in } Z_N$$
$$= |\{x : GCD(x, N) = 1\}| .$$

Euler's Theorem

$$\forall N \in \mathbb{N}, \forall x \in Z_N^*,$$
$$x^{\phi(N)} = 1 \text{ in } Z_N .$$

Public Key Cryptography



RSA Cryptosystem

Alice generates

- $N = pq$

RSA Cryptosystem

Alice generates

- $N = pq$
- $e \cdot d = 1 \mod \phi(N)$

RSA Cryptosystem

Alice generates

- $N = pq$
- $e \cdot d = 1 \mod \phi(N)$
- $\mathsf{pk} = (N, e)$

RSA Cryptosystem

Alice generates

- $N = pq$
- $e \cdot d = 1 \mod \phi(N)$
- $\mathsf{pk} = (N, e)$
- $\mathsf{sk} = (N, d)$

RSA Cryptosystem

Alice generates

- $N = pq$
- $e \cdot d = 1 \mod \phi(N)$
- $\mathsf{pk} = (N, e)$
- $\mathsf{sk} = (N, d)$

Encryption/Decryption

For a message $m \in \mathbb{Z}_N^*$:

$$c = \mathsf{Enc}(pk, m) = \mathsf{Enc}(N, e, m) = m^e \text{ in } Z_N^* .$$

RSA Cryptosystem

Alice generates

- $N = pq$
- $e \cdot d = 1 \mod \phi(N)$
- $\mathsf{pk} = (N, e)$
- $\mathsf{sk} = (N, d)$

Encryption/Decryption

For a message $m \in \mathbb{Z}_N^*$:

$$c = \mathsf{Enc}(pk, m) = \mathsf{Enc}(N, e, m) = m^e \text{ in } Z_N^* .$$

For a ciphertext $c \in \mathbb{Z}_N^*$:

$$m = \mathsf{Dec}(sk, c) = \mathsf{Dec}(N, d, c) = c^d \text{ in } Z_N^* .$$

Fast, Correct, Secure

Ubiquitous RSA

- Online banking
- SSL/TLS
- Emails
- Secure file systems
- ...

Attacks on (bad implementations of) RSA

Textbook RSA is not Secure

Factoring and RSA

RSA with Prime Modulus

Small Difference

Not Enough Randomness

PKCS1

Summary

- Encryption must be randomized!

Summary

- Encryption must be randomized!
- RSA is powerful and ubiquitous

Summary

- Encryption must be randomized!
- RSA is powerful and ubiquitous
- Simple, but needs to be implemented correctly

Summary

- Encryption must be randomized!
- RSA is powerful and ubiquitous
- Simple, but needs to be implemented correctly
- There are many great implementations