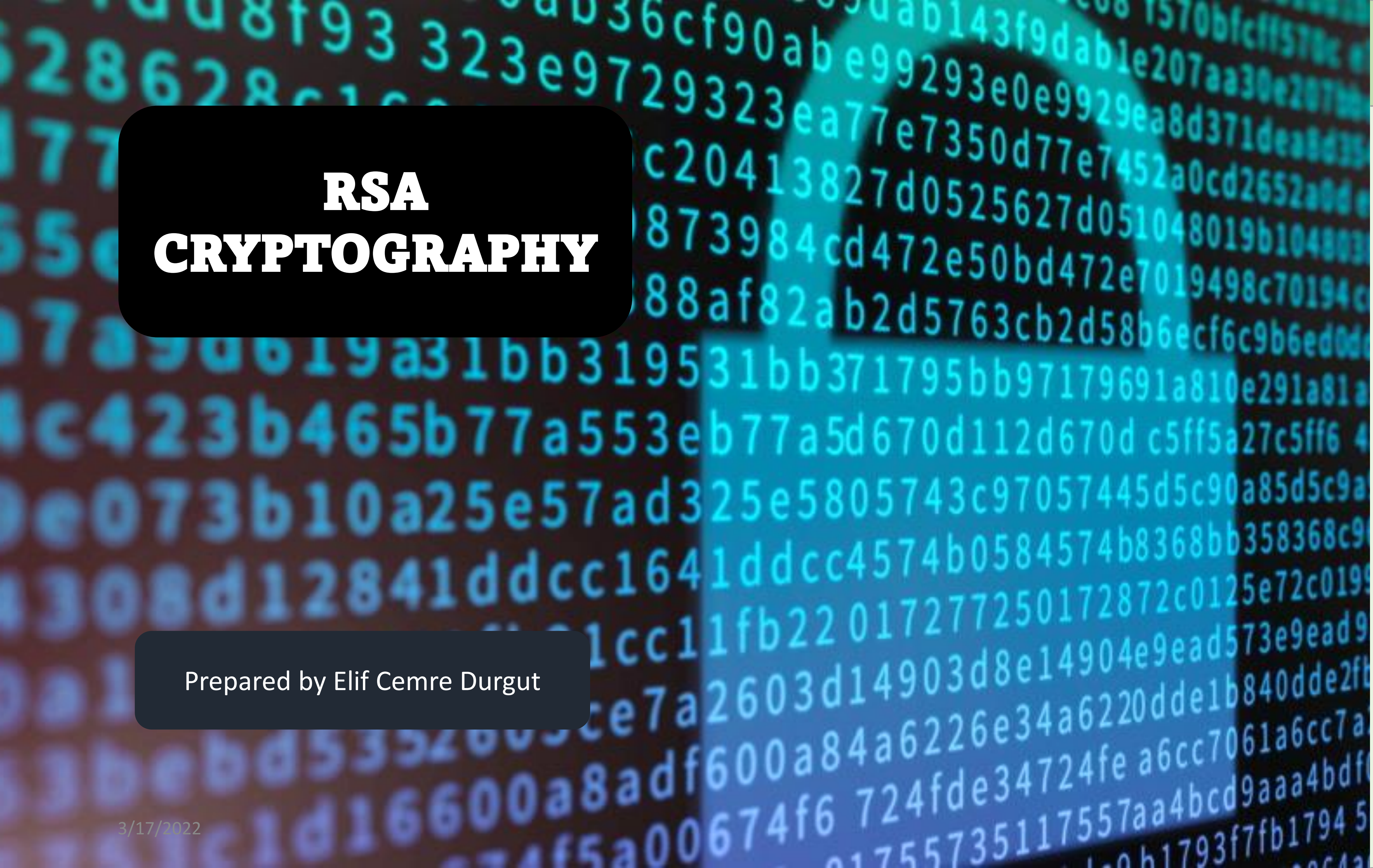
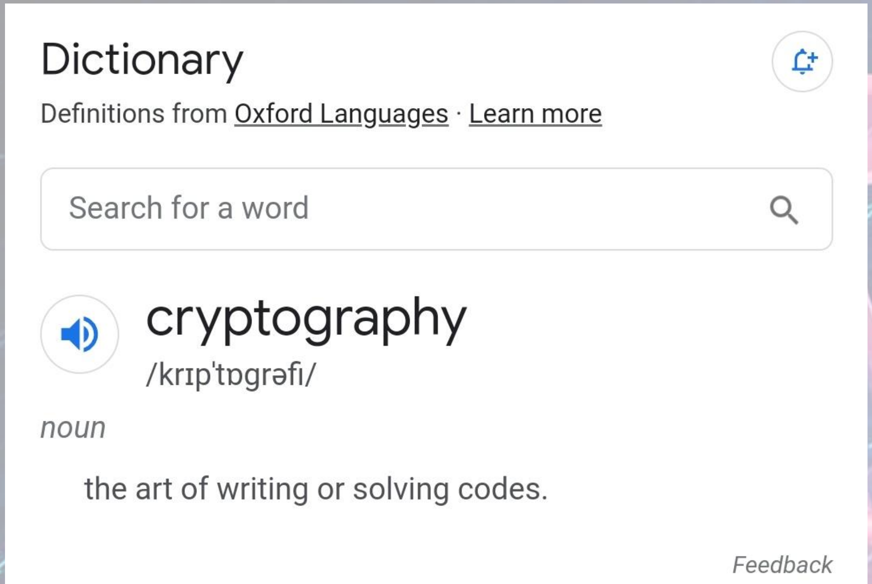


RSA CRYPTOGRAPHY

Prepared by Elif Cemre Durgut

3/17/2022

RSA CRYPTOGRAPHY


Dictionary

Definitions from Oxford Languages · Learn more

Search for a word

cryptography

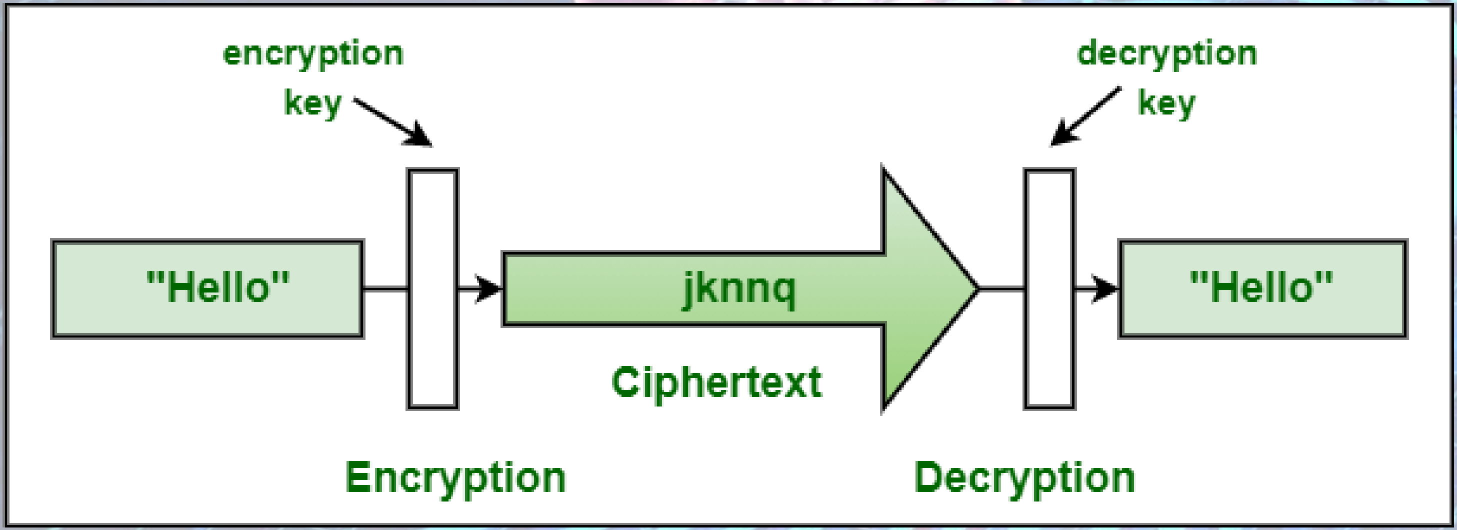
/krɪpˈtɒɡrəfi/

noun

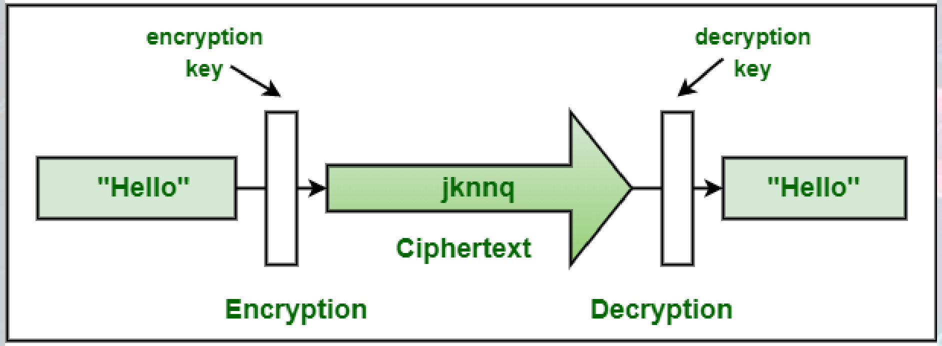
the art of writing or solving codes.

Feedback

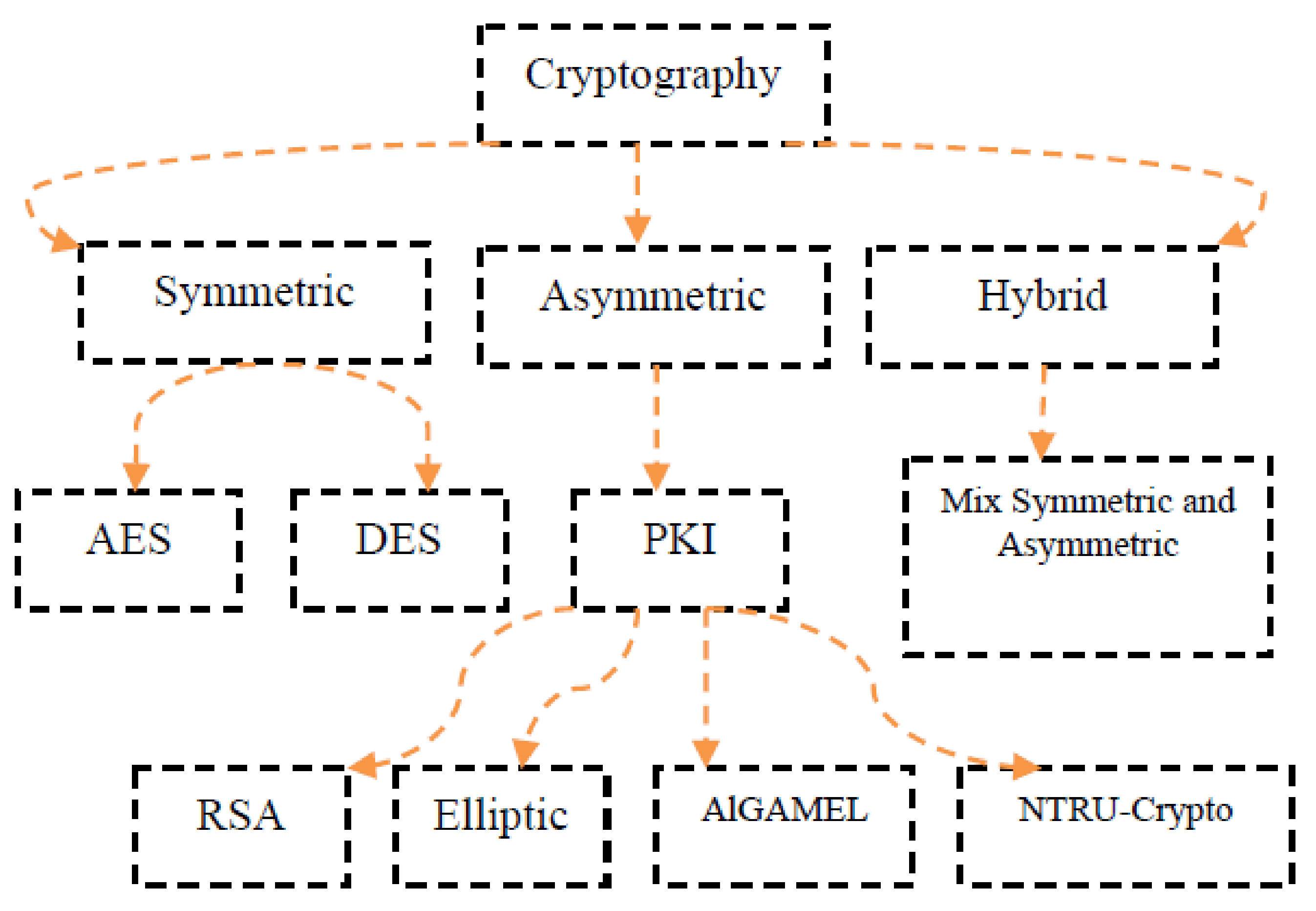
Secure communication techniques using mathematical algorithms

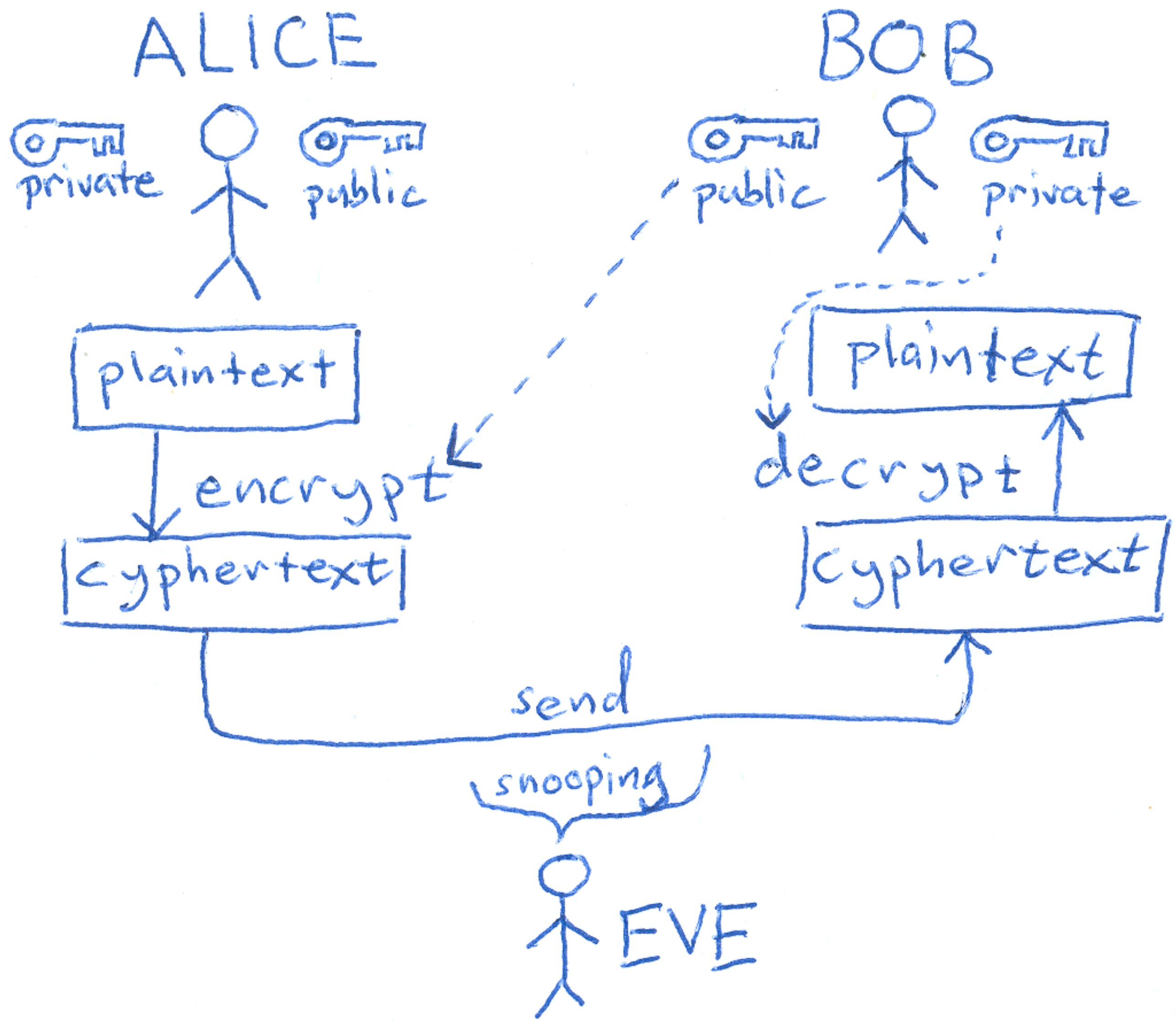


3/17/2022



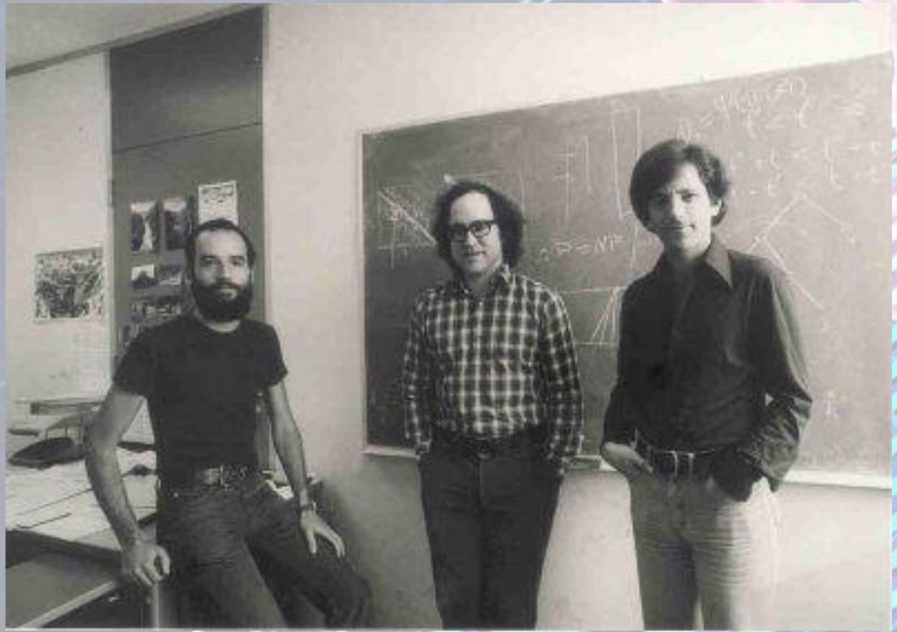
Cryptography




ALICE
BOB
private
public
public
private
plaintext
plaintext
encrypt
decrypt
cyphertext
cyphertext
send
snooping
EVE

Rivest
Shamir
Adleman

~ one of the most popular public key cryptosystems
~ Introduced by Rivest, Shamir and Adleman in 1977 at MIT
~ RSA patent expired in 2000 → can be used for commercial and non-commercial use for free
~ Based on Integer Factorization problem
~ Each user has public and private key pair



How does RSA work?

1) Choose two large prime numbers p and q (2048 bit)

2) Compute $n = p \times q$

3) Compute $\boldsymbol{\Phi(n) = (p-1)(q-1)}$

4) Choose a random integer, $\boldsymbol{0 < e < \Phi(n)}$ with $\boldsymbol{\gcd(e, \Phi(n)) = 1}$

5) Compute the inverse $\boldsymbol{d = e^{-1} \bmod \Phi(n)}$,

i.e. $e \cdot d \equiv 1 \bmod \Phi(n)$

(Note: This will help us in the proof!)

Public keys: e, n
Private keys: p, q and d

Encryption is done using the receiver's public key:

$$y \equiv x^e \bmod n, \quad \text{where } x < n$$

Decryption is done using the receiver's private key:

$$x \equiv y^d \bmod n$$

EXAMPLE

If Alice wants to send a message to Bob, she needs Bob's public key

ALICE	BOB
	Chooses $p = 3$ and $q = 11$
	$N = p \cdot q = 33$
	$\Phi(n) = (p-1)(q-1) = 20$
	Chooses $e = 3$
	$d = e^{-1} \bmod \Phi(n) = 7$
	Sends (e, n) to Alice.
Message $x = 4$	
Encrypts $y \equiv x^e \bmod n = 31$	
Sends y to Bob.	
	Decrypts $x \equiv y^d \bmod n = 4$

WHY DOES RSA WORK?

E → $y \equiv x^e \bmod n$
D → $x \equiv y^d \bmod n$

We want to prove $y^d \bmod n \equiv x$

$y^d \bmod n \equiv (x^e \bmod n)^d \bmod n$

$\equiv x^{ed} \bmod n$

$e \cdot d \equiv 1 \bmod \Phi(n)$ → $e \cdot d = 1 + k \cdot \Phi(n)$ (k: non-negative integer)

$x^{ed} \bmod n \equiv x^{1 + k\Phi(n)} \bmod n$

$\equiv x^1 \cdot x^{k\Phi(n)} \bmod n$

If $\gcd(x, n) = 1$, the proof ends here.

Euler's Theorem
$\gcd(x, n) = 1$ → $x^{\Phi(n)} \equiv 1 \bmod n$

Theoretical complexity classes

There is also a theoretical hierarchy of complexity classes refering to families of computational problems and the complexity of an optimal algorithm for solving these problems.

- **P** is the class of all problems for which the output is a boolean (yes/no) and where there is a polynomial time algorithm.
- **NP** is the class of problems with a boolean output for which the verification problem (checking that the answer is correct) is in **P**. Examples of an **NP** problem is asking whether a system of inequalities has a solution. If there is a solution the claimant can give the solution which can be verified in polynomial time.

Theoretical complexity classes

MAA507: Computational complexity
Christopher Engström
Big O notations
Complexity analysis
Types of complexity
Complexity classes

- **NP-complete** is the class of problems for which any other **NP** problem can be transformed into an instance of that problem, suprisingly most **NP** problems with no known polynomial algorithm are **NP-complete**. In some way they are all "equally difficult" since you can easily go from an instance of one problem into another.
- All known algorithms for **NP-complete** problems have exponential complexity.
- It is not a severe restriction that **NP** problems only answer yes/no questions. A bit string of any length can be calculated through a series of yes/no questions whether the kth bit is 1.

Why is it hard to break RSA?

~ RSA depends on the hardness of the integer factorization problem.

~ Integer factorization is in NP but not NP-complete.

~ There are no efficient known factoring algorithms for large integers.

~ Largest integer factored so far is 829-bit.

~ The best is General Number Field Sieve which has sub-exponential complexity

Mathematical Attacks

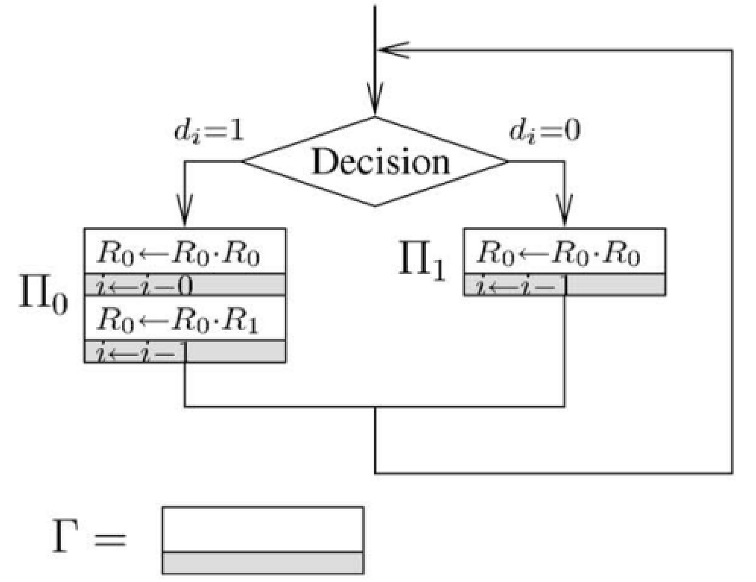
1) Factoring N

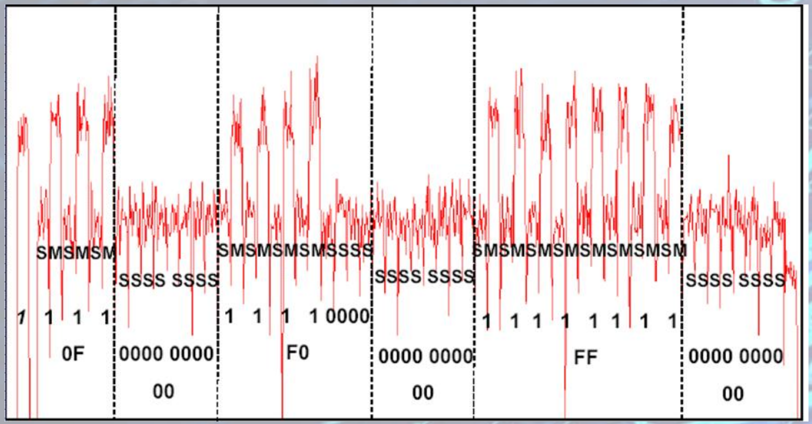
Side Channel Attacks

1) Simple P. A.
2) Differential P. A.
3) Timing Attacks

Input: $x, d = (d_{m-1}, \ldots, d_0)_2$
Output: $y = x^d$

$R_0 \leftarrow 1\,;\, R_1 \leftarrow x\,;\, i \leftarrow m-1$
while $(i \geq 0)$ **do**
  $R_0 \leftarrow (R_0)^2$
  if $(d_i = 1)$ **then** $R_0 \leftarrow R_0 \cdot R_1$
  $i \leftarrow i - 1$
endwhile
return R_0



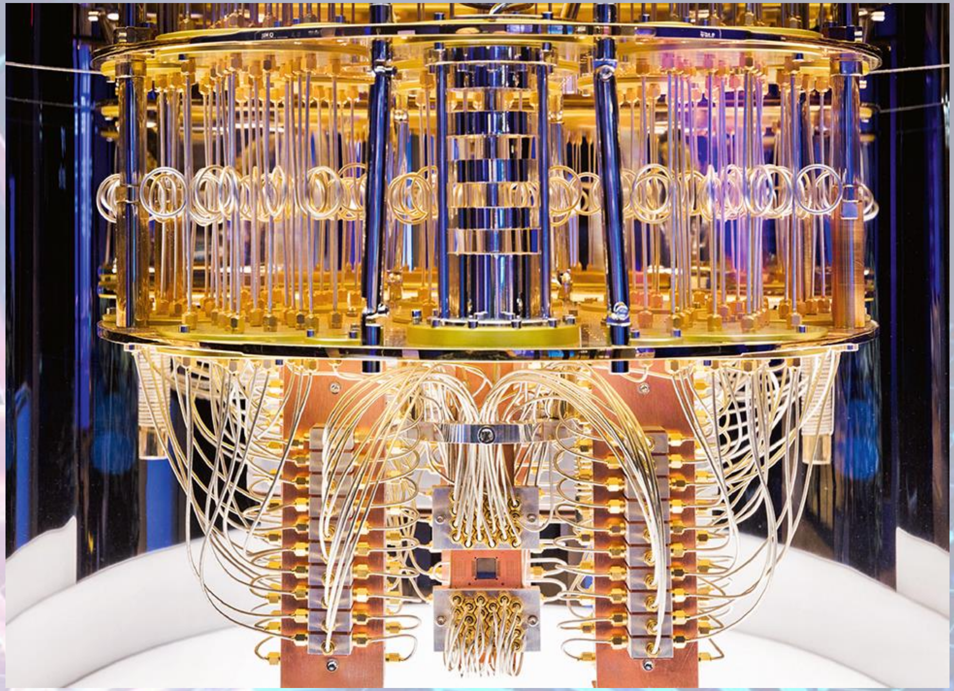


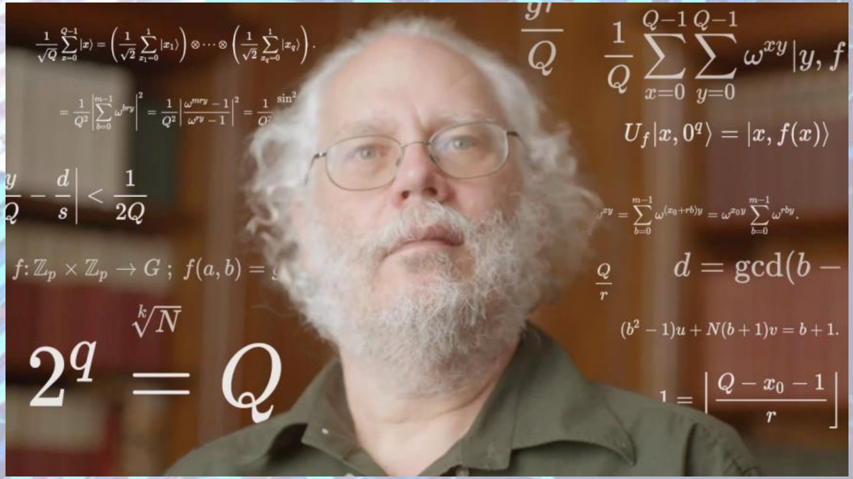
What about Quantum computers?

~Shor's Algorithm in 1994 by Peter Shor

~Integer factorization can be done in polynomial time using quantum computers

~The largest integer factored is 21?





RSA CRYPTOGRAPHY

What is cryptography?

RSA

RSA Setup - Algorithm

Why does it work?

Hardness of RSA

Attacks to RSA

Future – Quantum Computers

REFERENCES

Bellare, M., Rogaway, P. (1995). Optimal Asymmetric Encryption How to Encrypt with RSA. https://cseweb.ucsd.edu/~mihir/papers/oaep.pdf

Bernstein, D. J., Heninger, N., Lou, P., Valenta, L. (2017). Post-quantum RSA. https://cr.yp.to/papers/pqrsa-20170419.pdf

Chevallier-Mames, B., Ciet, M., Joye, M. (2004). Low-Cost Solutions for Preventing Simple Side-Channel Analysis: Side-Channel Atomicity. *IEEE Transactions on Computers, Vol. 53, No. 6.* https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.457.2254&rep=rep1&type=pdf

Courrege, J., Feix, B., Rousselet, M. (2010). Simple Power Analysis on Exponentiation Revisited. *IFIP International Federation for Information Processing.* https://link.springer.com/content/pdf/10.1007%2F978-3-642-12510-2_6.pdf

Kocher, P. (1996). Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. https://paulkocher.com/doc/TimingAttacks.pdf

Kocher, P., Jaffe, J., Jun, B. (1994). Differential Power Analysis. https://www.paulkocher.com/doc/DifferentialPowerAnalysis.pdf

Rivest, R. L., Shamir, A., Adleman, L. (1977). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. https://people.csail.mit.edu/rivest/Rsapaper.pdf

Zhou, Y., Feng, D. (2005). Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing. *National Natural Science Foundation of P.R. China.* https://csrc.nist.gov/csrc/media/events/physical-security-testing-workshop/documents/papers/physecpaper19.pdf