EXCALIBUR

Protocolos de generación de llaves para una desencriptación jerárquica DAG con múltiples participantes

Geraldine Monsalve

INTRODUCCIÓN

INTRODUCCIÓN

¿Qué es la criptografía?

INTRODUCCIÓN

¿Qué es la criptografía?

> [. . .] involves the study of mathematical techniques for securing digital information, systems, and distributed computations against adversarial attacks. (Jonathan Katz)

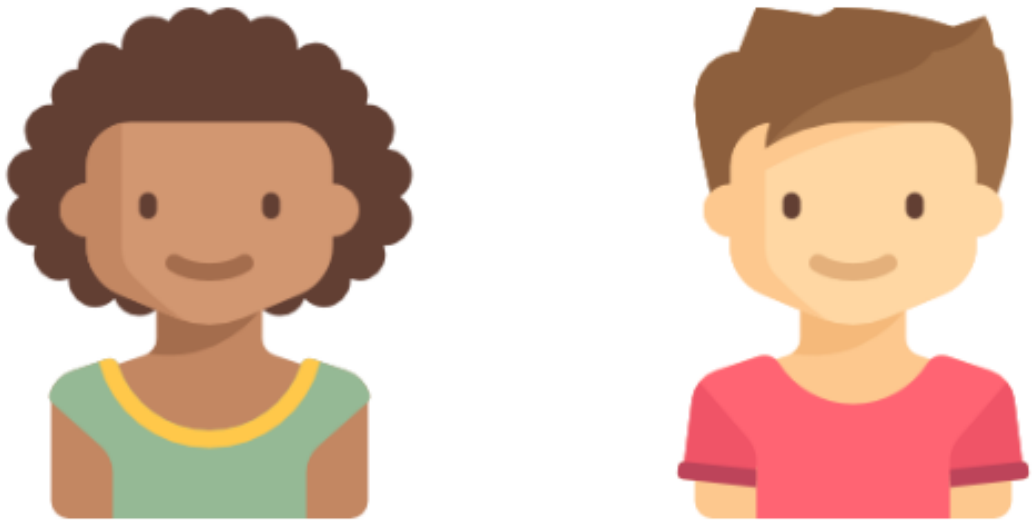


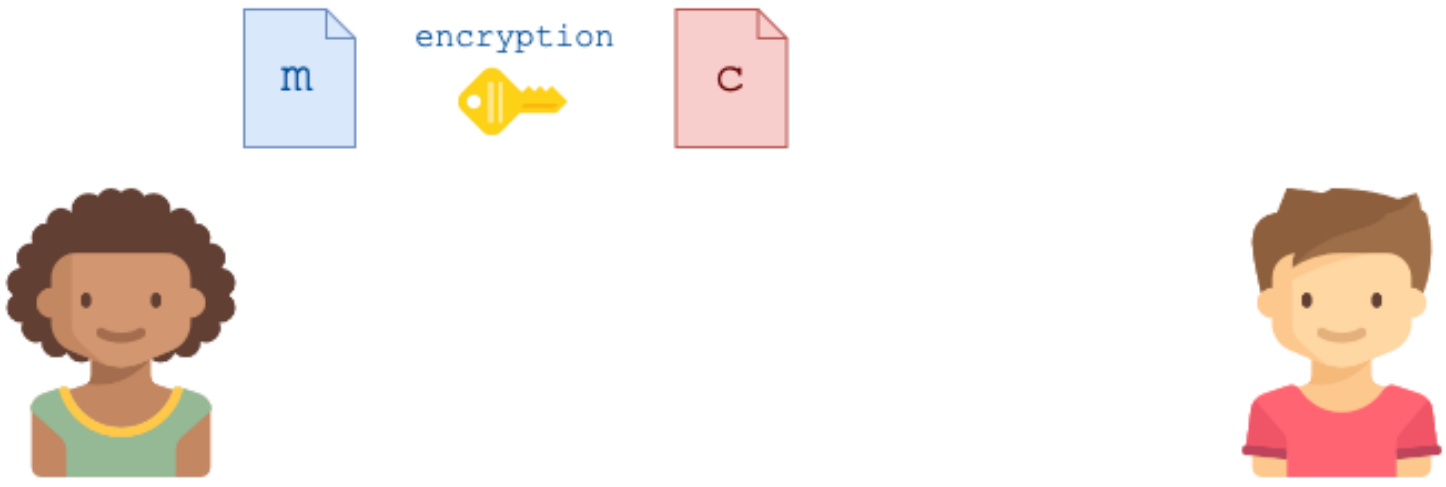
Figure: Alice y Bob

Figure: y Eve

PRIVATE-KEY SETTING

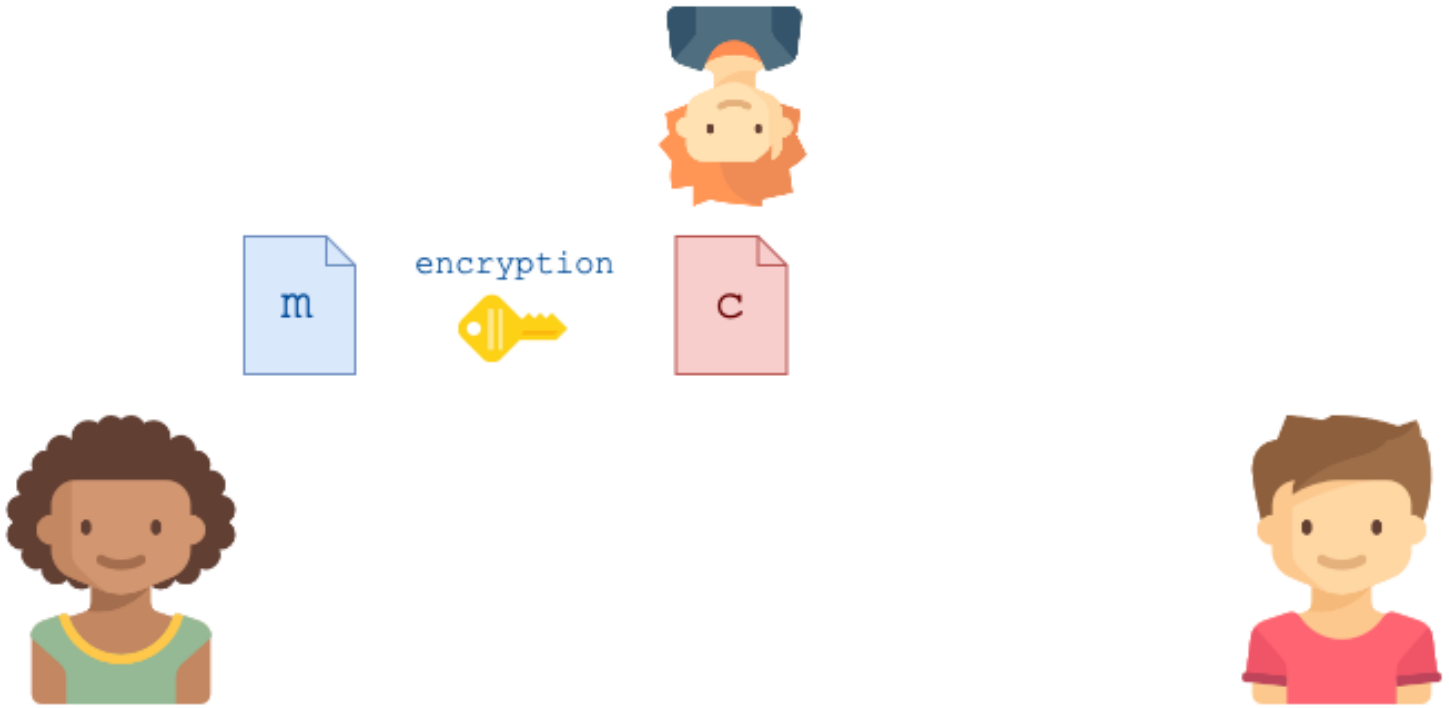
Alice quiere enviar un mensaje a Bob. Ambos comparten un secreto: su llave privada.

PRIVATE-KEY SETTING



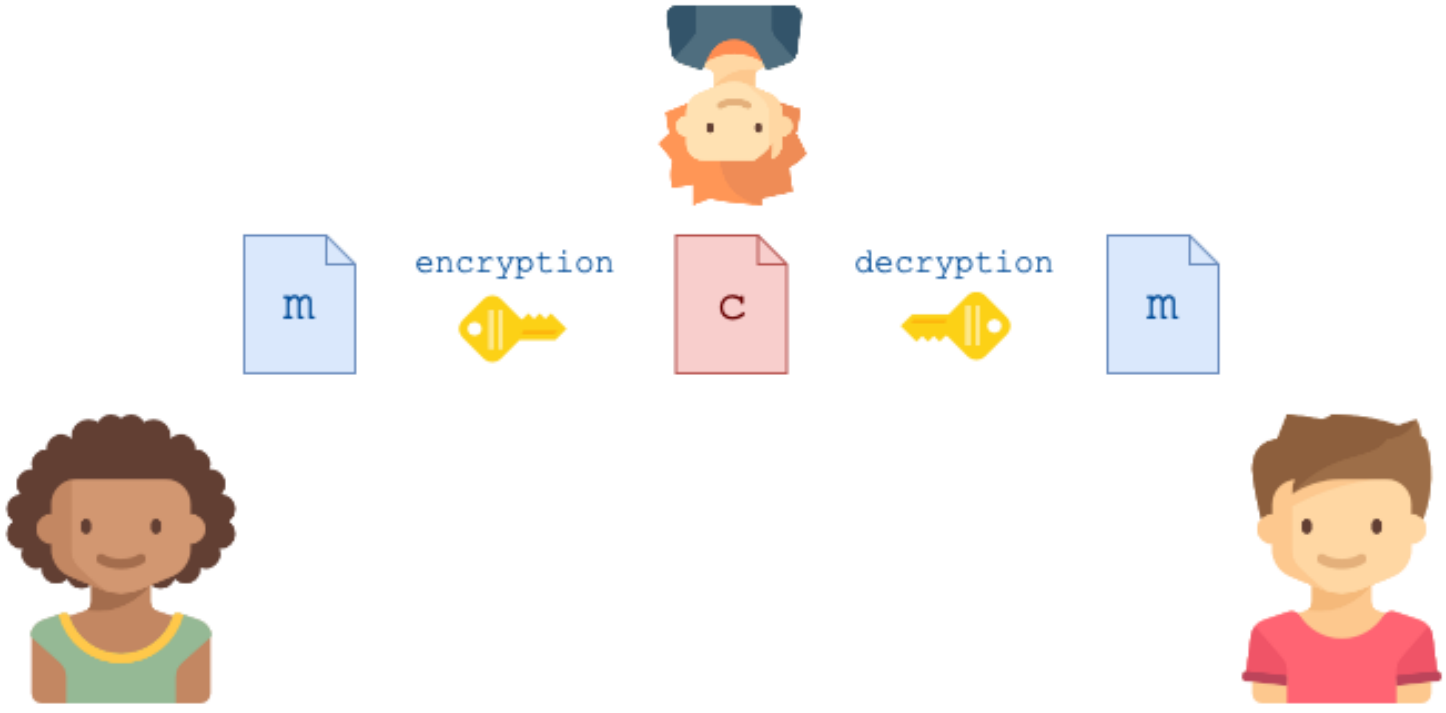
Alice cifra el mensaje con la llave privada compartida y luego lo envía a Bob.

PRIVATE-KEY SETTING



Eve puede monitorear el canal de comunicación.

PRIVATE-KEY SETTING



Bob desencripta el mensaje cifrado con la llave compartida y obtiene el mensaje. original

PRIVATE-KEY SETTING

- Gen : Entrega una llave de $\mathcal{K}$, obtenida de manera uniformemente aleatoria.
- Enc : Dada una llave k $\in \mathcal{K}$ y un mensaje m $\in \mathcal{M}$, entrega un mensaje cifrado c $\in \mathcal{C}$.
- Dec : Dada una llave k $\in \mathcal{K}$ y un mensaje cifrado c $\in \mathcal{C}$, entrega un texto plano m $\in \mathcal{M}$.

PRIVATE-KEY SETTING

- Gen : Entrega una llave de $\mathcal{K}$, obtenida de manera uniformemente aleatoria.
- Enc : Dada una llave $k \in \mathcal{K}$ y un mensaje $m \in \mathcal{M}$, entrega un mensaje cifrado $c \in \mathcal{C}$.
- Dec : Dada una llave $k \in \mathcal{K}$ y un mensaje cifrado $c \in \mathcal{C}$, entrega un texto plano $m \in \mathcal{M}$.

El esquema debe cumplir $\mathrm{Dec}(k, \mathrm{Enc}(k, m)) = m$.

PUBLIC-KEY SETTING

Llave pública

Llave privada

Alice genera un par de llaves: una pública pk_A y una privada sk_A

PUBLIC-KEY SETTING

Bob quiere enviar un mensaje a Alice y lo encripta con la llave pública pk_A

PUBLIC-KEY SETTING

Alice descrencripta el mensaje cifrado con su llave privada sk_A

PUBLIC-KEY SETTING

- Gen : Entrega un par de llaves de $\mathcal{K}$, obtenidas de manera uniformemente aleatoria.
- Enc : Dada una llave pública pk $\in \mathcal{K}$ y un mensaje m $\in \mathcal{M}$, entrega un mensaje cifrado c $\in \mathcal{C}$.
- Dec : Dada una llave privada sk $\in \mathcal{K}$ y un mensaje cifrado c $\in \mathcal{C}$, entrega un texto plano m $\in \mathcal{M}$.

PUBLIC-KEY SETTING

- Gen : Entrega un par de llaves de $\mathcal{K}$, obtenidas de manera uniformemente aleatoria.
- Enc : Dada una llave pública pk $\in \mathcal{K}$ y un mensaje m $\in \mathcal{M}$, entrega un mensaje cifrado c $\in \mathcal{C}$.
- Dec : Dada una llave privada sk $\in \mathcal{K}$ y un mensaje cifrado c $\in \mathcal{C}$, entrega un texto plano m $\in \mathcal{M}$.

El esquema debe cumplir $\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{m})) = \mathsf{m}$.

MOTIVACIÓN

MOTIVACIÓN

Protocolos de generación de llaves para una desencriptación jerárquica DAG con múltiples participantes

MOTIVACIÓN

Protocolos de generación de llaves para una desencriptación jerárquica DAG con múltiples participantes

MOTIVACIÓN

Protocolos de generación de llaves para una desencriptación jerárquica DAG con **múltiples participantes**

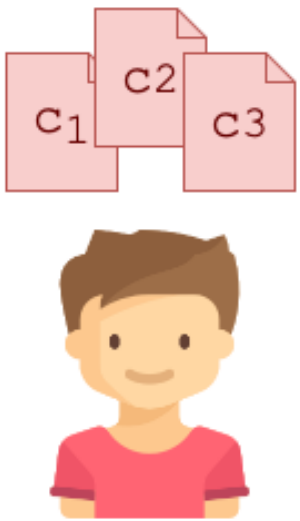
MOTIVACIÓN

Protocolos de generación de llaves para una **desencriptación jerárquica DAG** con múltiples participantes

DESENCRIPTACIÓN JERÁRQUICA

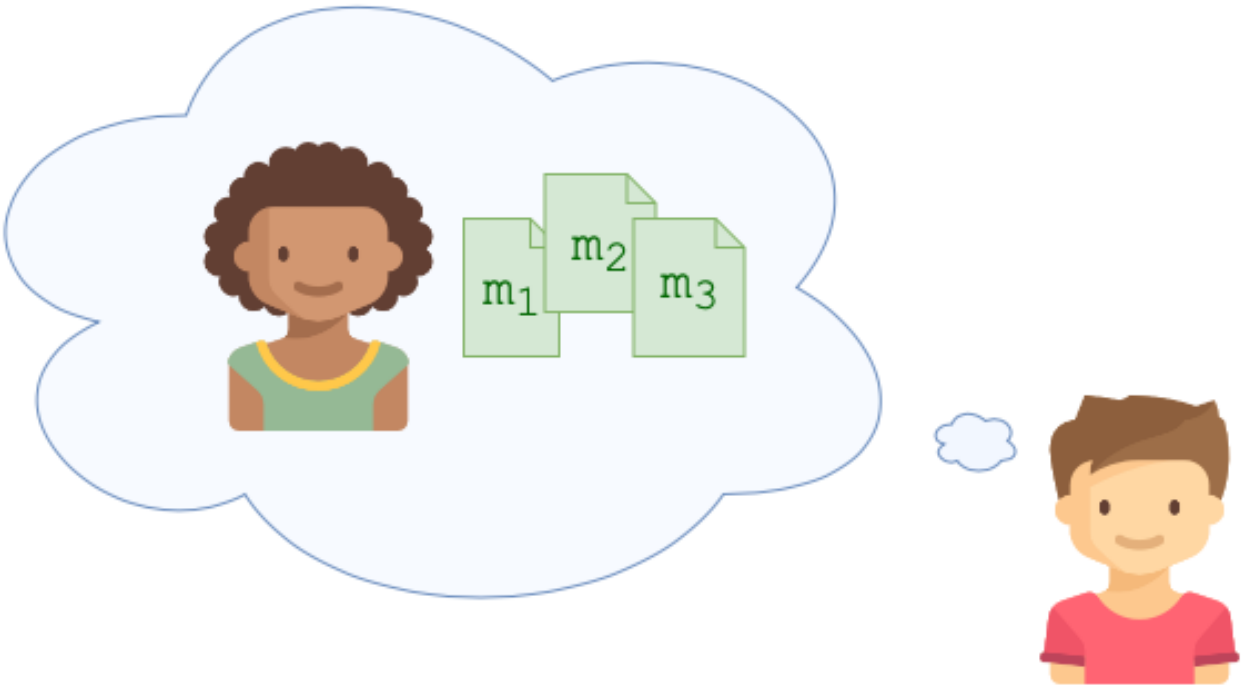
Bob es empleado de Alice

DESENCRIPTACIÓN JERÁRQUICA



Bob recibe mensajes cifrados dirigidos a él, es decir, encriptados con su llave pública pk_B

DESENCRIPTACIÓN JERÁRQUICA



A Bob le gustaría que Alice pueda desencriptar sus mensajes sin necesidad de revelar su llave secreta pk_B

DAG: DIRECTED ACYCLIC GRAPH

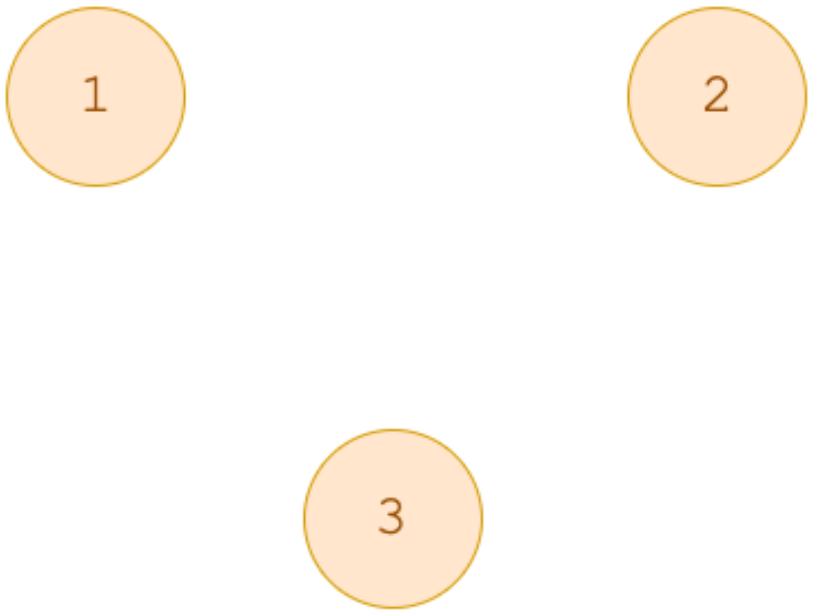


Figure: Grafo: nodos

DAG: DIRECTED ACYCLIC GRAPH

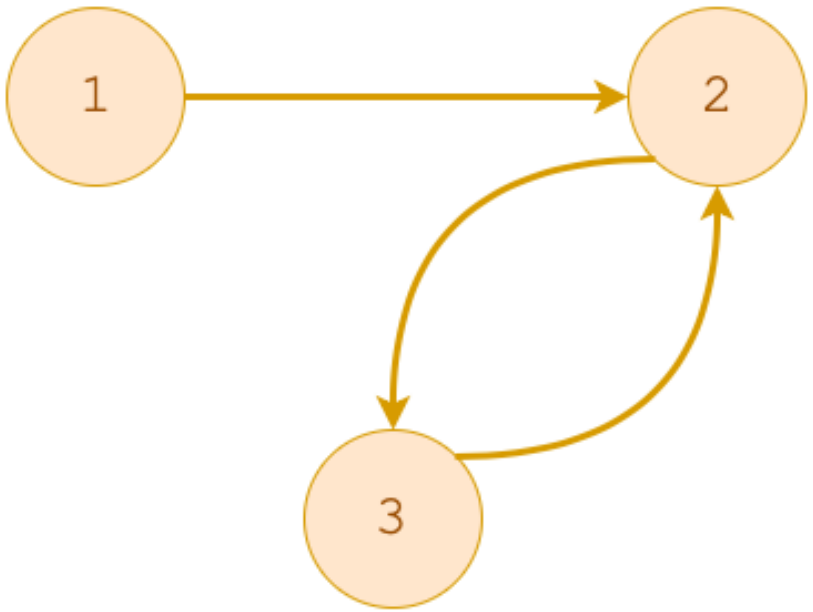


Figure: Grafo: aristas

DAG: DIRECTED ACYCLIC GRAPH

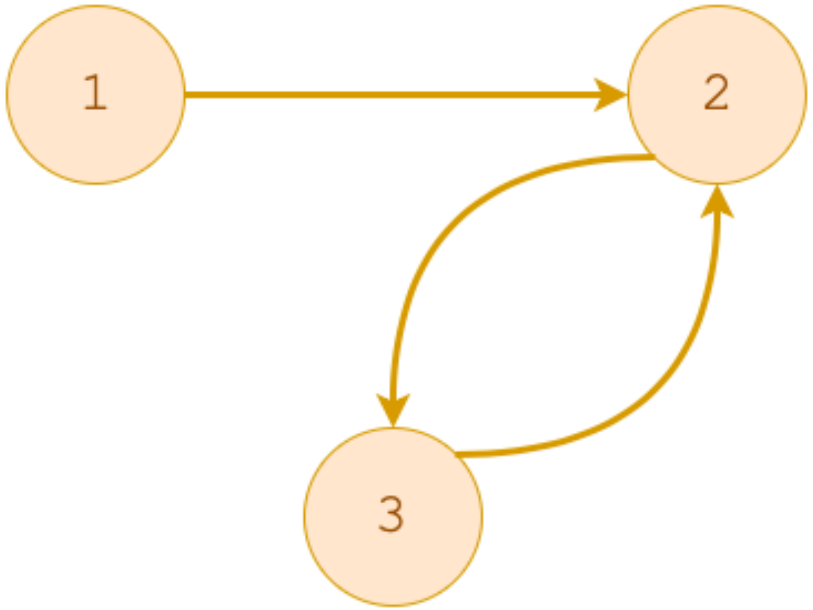


Figure: Grafo cíclico

DAG: DIRECTED ACYCLIC GRAPH

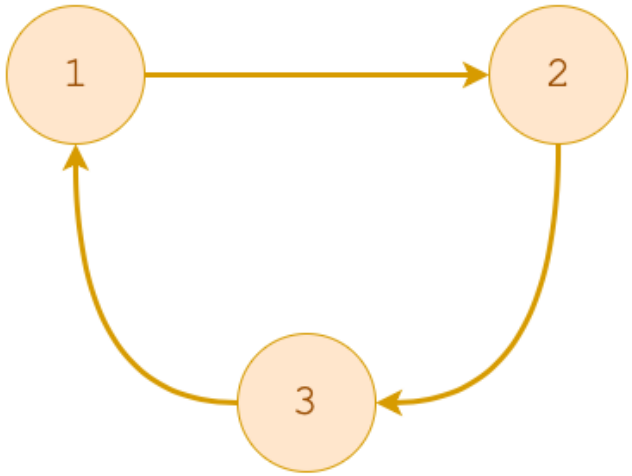


Figure: Grafo cíclico

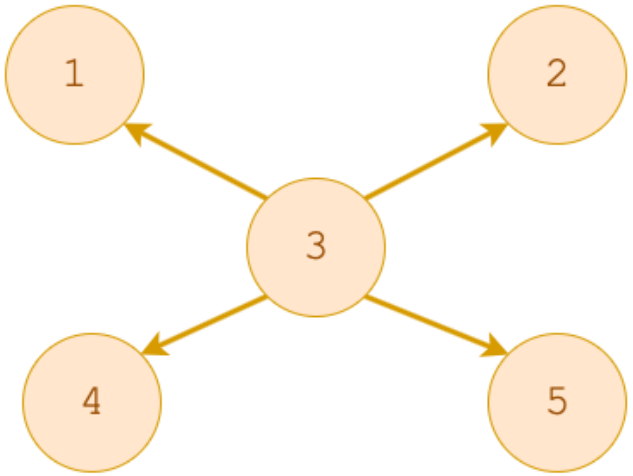


Figure: Grafo acíclico

MOTIVACIÓN

¿Cómo se ve la configuración nos interesa?

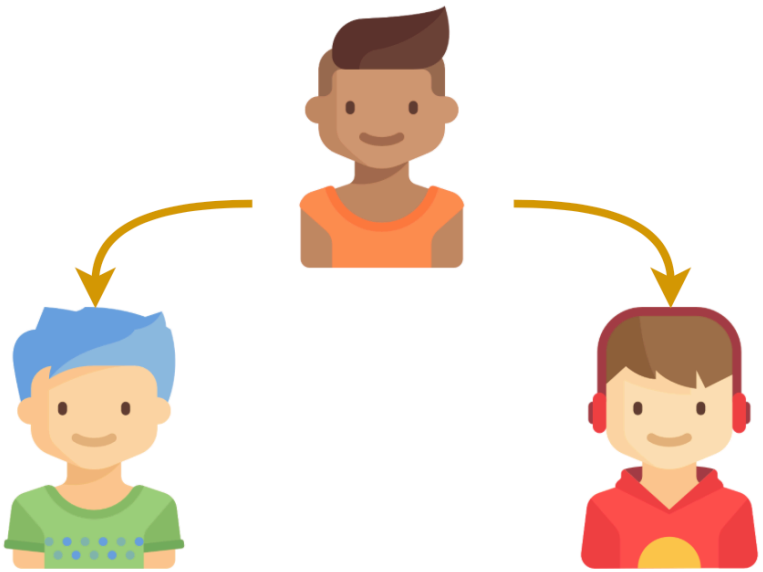


Figure: Desencriptación jerárquica DAG

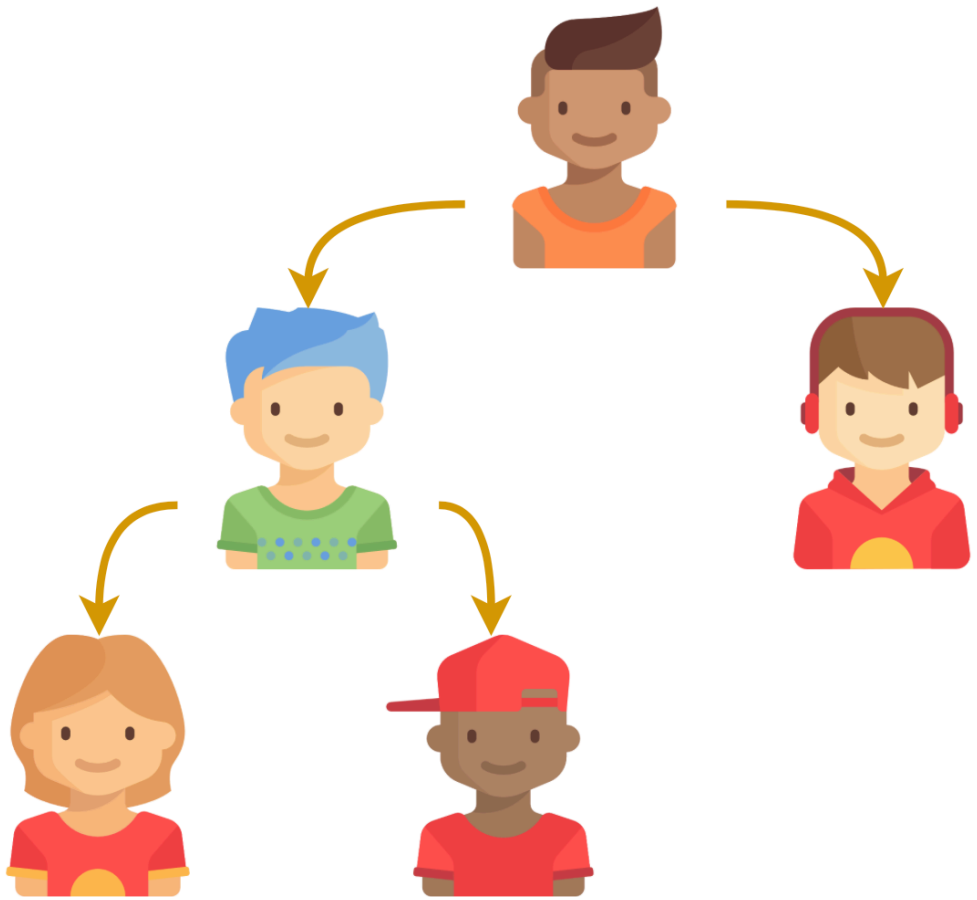


Figure: Desencriptación jerárquica DAG

PROCESO

TEMA DE TESIS

Acá van tres ideas de investigación que podrían interesarle a colegas o alumnos:

TEMA DE TESIS

Acá van tres ideas de investigación que podrían interesarle a colegas o alumnos:

1. 密碼分析：猜測 中的平行六面體（自適應版本)
2. 多個參與者的Excalibur協議
3. 考慮漢明假設

Blending FHE-NTRU keys – The Excalibur Property*

Louis Goubin and Francisco José Vial Prado

Laboratoire de Mathématiques de Versailles
UVSQ, CNRS, Université Paris-Saclay
78035 Versailles, France

May 2, 2017

Abstract

Can Bob give Alice his decryption secret and be convinced that she will not give it to someone else? This is achieved by a proxy re-encryption scheme where Alice does not have Bob's secret but instead she can transform ciphertexts in order to decrypt them with her own key. In this article, we answer this question in a different perspective, relying on a property that can be found in the well-known modified NTRU encryption scheme. We show how parties can collaborate to *one-way-glue* their secret-keys together, giving Alice's secret-key the additional ability to decrypt Bob's ciphertexts. The main advantage is that the protocols we propose can be plugged directly to the modified NTRU scheme with no post-key-generation space or time costs, nor any modification of ciphertexts. In addition, this property translates to the NTRU-based multikey homomorphic scheme, allowing to equip a hierarchic chain of users with automatic re-encryption of messages and supporting homomorphic operations of ciphertexts. To achieve this, we propose two-party computation protocols in cyclotomic polynomial rings. We base the security in presence of various types of adversaries on the RLWE and DSPR assumptions, and on two new problems in the modified NTRU ring.

¿Qué resuelve el paper original?

¿Qué resuelve el paper original?

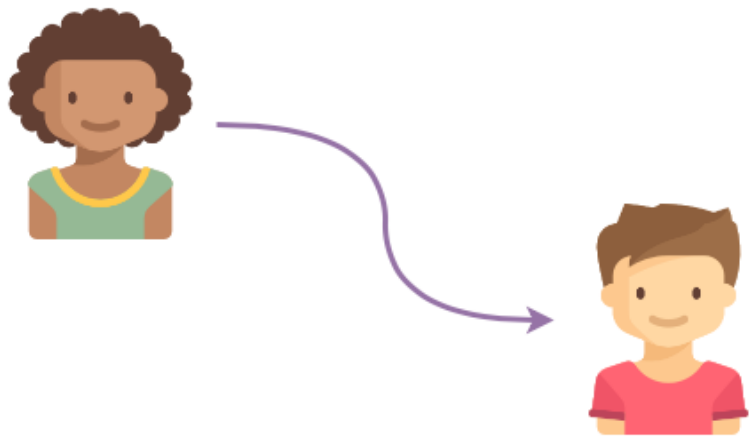


Figure: Desencriptación jerárquica original

¿Qué resuelve el paper original?

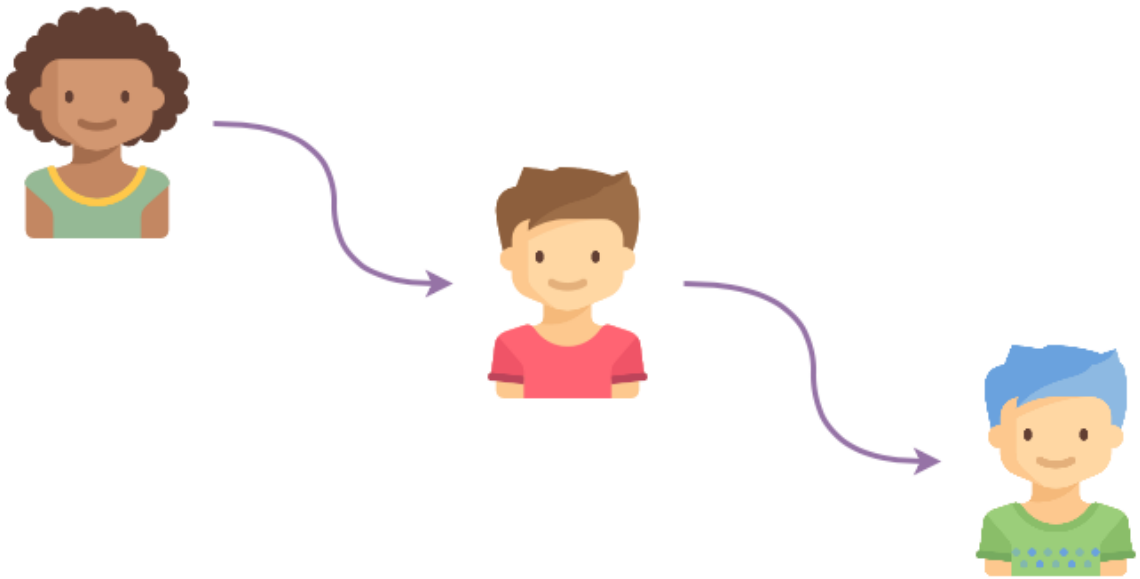


Figure: Desencriptación jerárquica original

THE EXCALIBUR PROPERTY

Un esquema de encriptación de llave pública $\mathcal{E} = (\text{Gen}, \text{Enc}, \text{Dec})$ posee la propiedad Excalibur si:

THE EXCALIBUR PROPERTY: 1

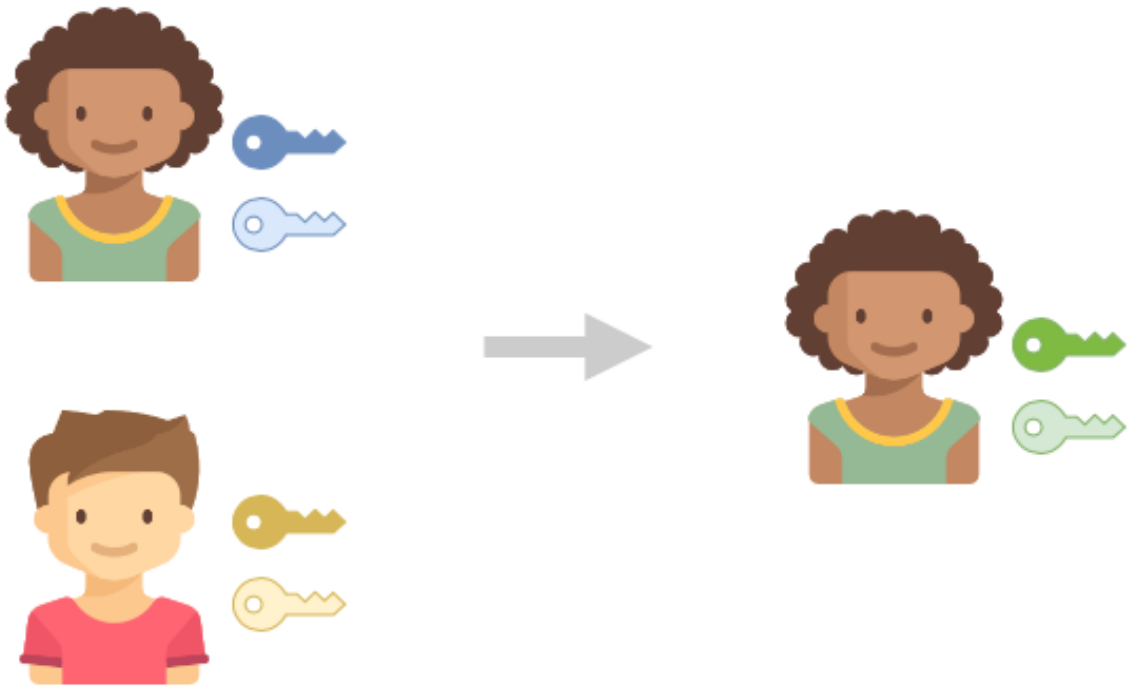


Figure: Algoritmo generación de llaves compuesta

THE EXCALIBUR PROPERTY: 2

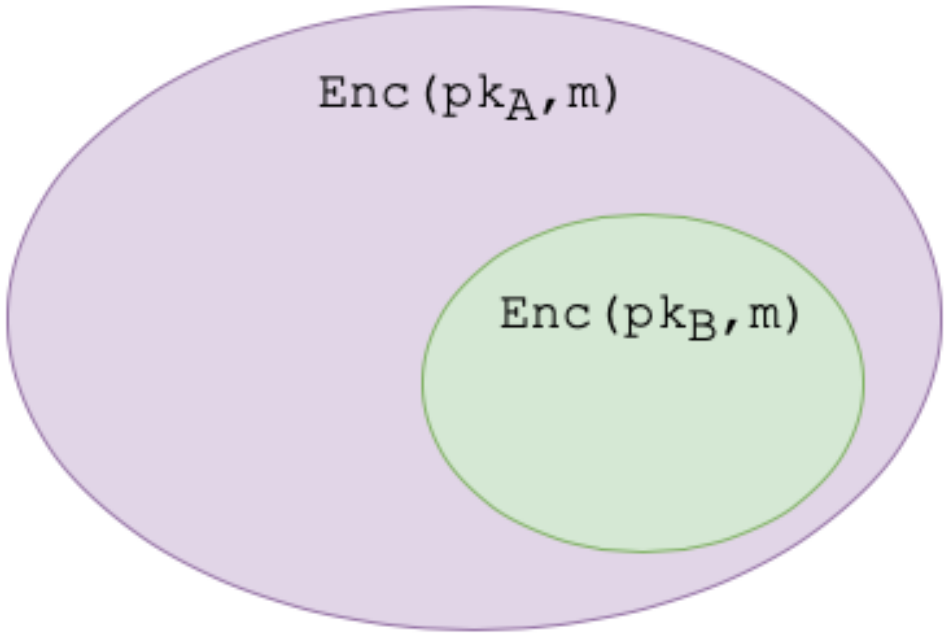


Figure: Alice puede desencriptar $c \in \text{Enc}(pk_A, \mathcal{M}) \cup \text{Enc}(pk_B, \mathcal{M})$

THE EXCALIBUR PROPERTY: 2

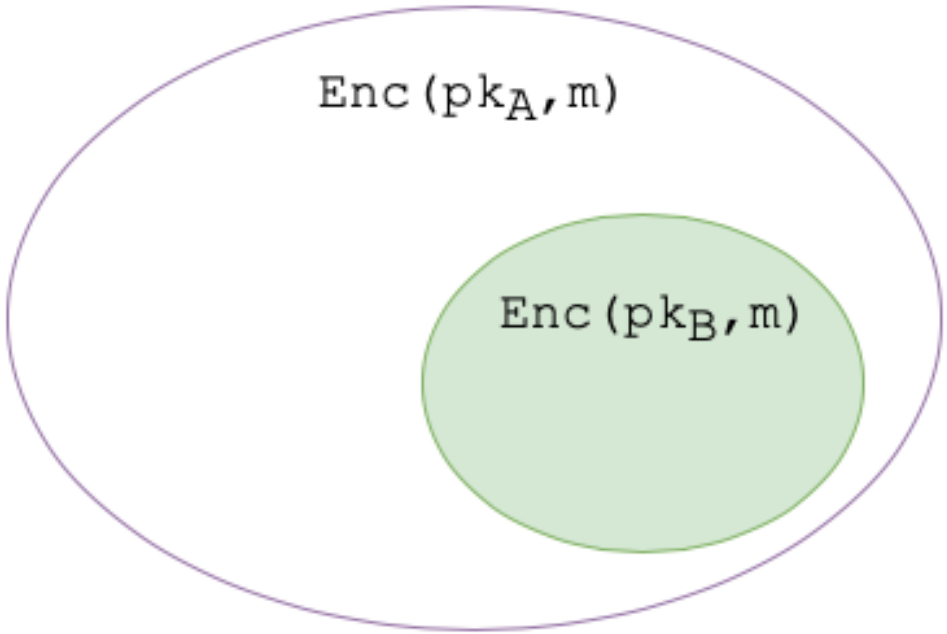


Figure: Bob no puede desencriptar $c \in \text{Enc}(\text{pk}_A, \mathcal{M})$

THE MULTIKEY PROPERTY

Figure: La nueva llave de Alice $\tilde{sk}_A = sk_A \cdot sk_B$

EXCALIBUR KEY GENERATION

¡Entonces solo necesitamos multiplicar un par de llaves!

EXCALIBUR KEY GENERATION

Ambas partes deben multiplicar los polinomios involucrados utilizando protocolos seguros de múltiples partes (SMPC), ya que no pueden confiar sus secretos al otro.

¿Qué hicimos nosotros?

¡Gracias!