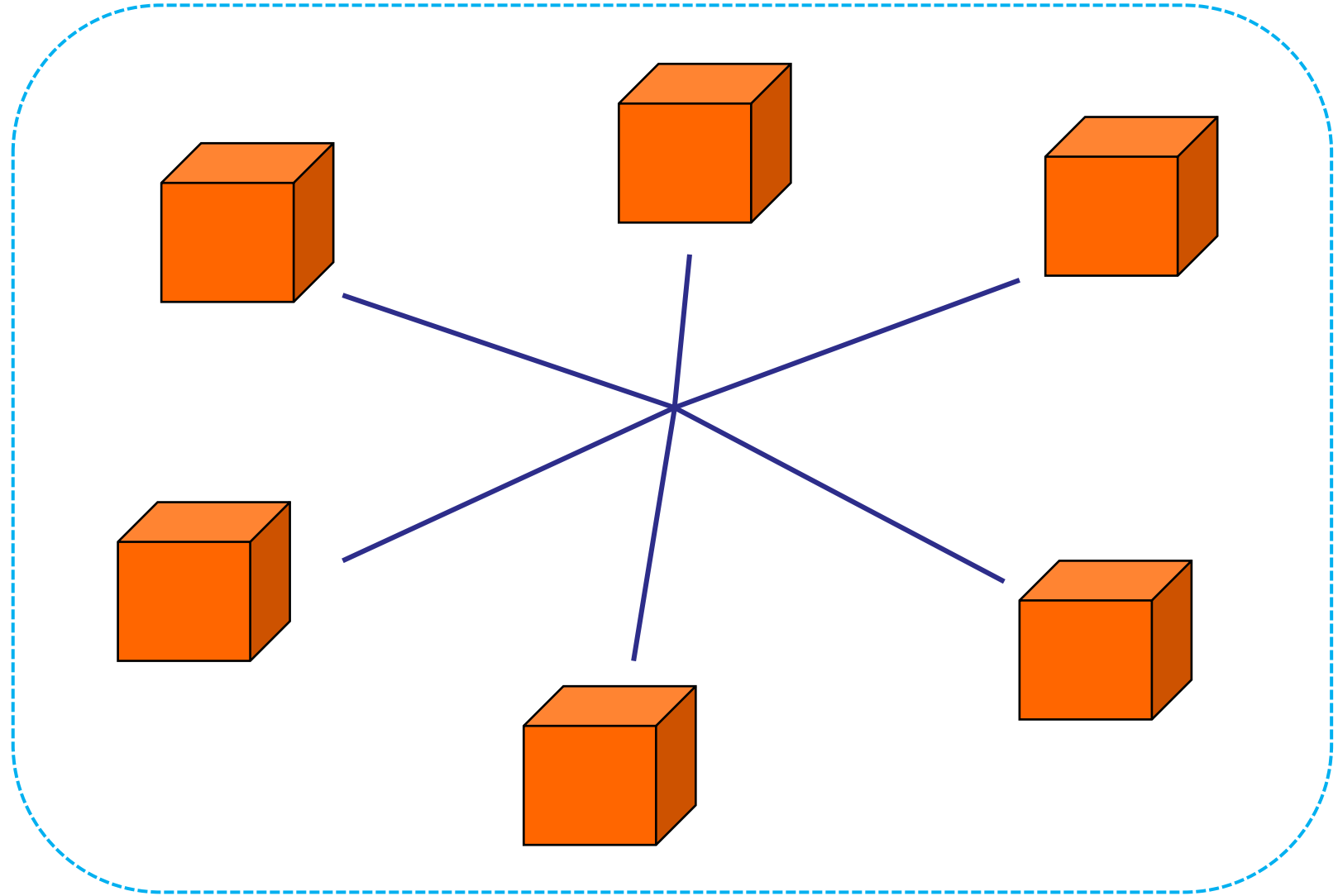


## Unidad N° 7:

# ADMINISTRACIÓN DE SEGURIDAD, TRANSACCIONES Y FALLOS EN SISTEMAS OPERATIVOS DISTRIBUIDOS



# SISTEMAS OPERATIVOS DISTRIBUIDOS

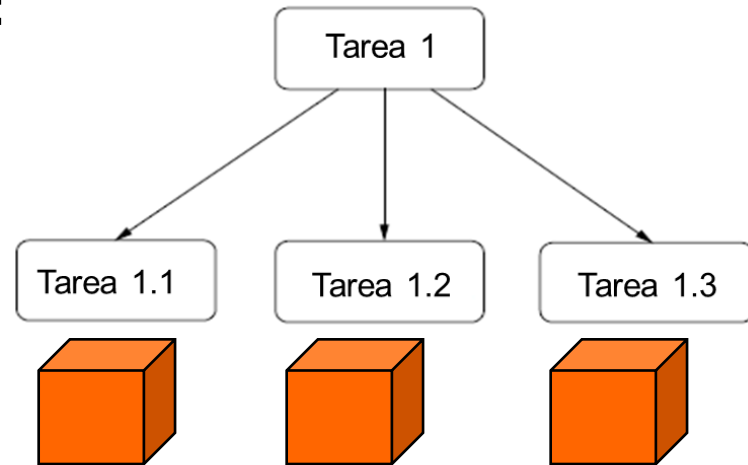


*Conjunto de computadoras que se integran para hacer desaparecer la dualidad local / remoto para ofrecer la visión de un «sistema único»*

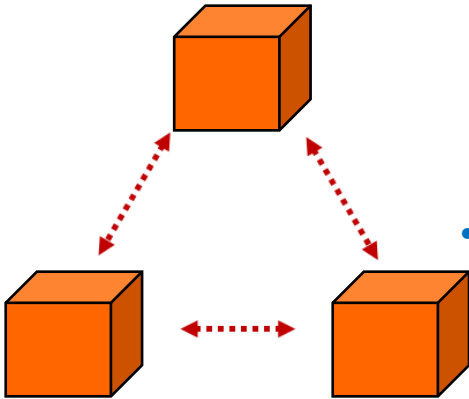
# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Objetivos de un Sistema Distribuido:

- Distribuir el Trabajo.



- Compartir Recursos.



- Logrando:
  - ✓ Alto Rendimiento
  - ✓ Alta Escalabilidad
  - ✓ Alta Disponibilidad

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Cuestiones para implementar un Sistema Distribuido:

- ❖ *¿Cómo distribuir la Carga de Trabajo?*
- ❖ *¿Cómo administrar los Recursos Compartidos?*
- ❖ *¿Cómo lograr la Sincronización de Procesos?*
- ❖ *¿Cómo manejar el Deadlock?*
- ❖ *¿Cómo lograr un ‘Estado Consistente’?*
- ❖ *¿Cómo asegurar la Confiabilidad y Fiabilidad?*

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ❖ *¿Cómo asegurar la Confiabilidad y Fiabilidad?*

→ Transacciones Distribuidas

→ Manejo de Fallos en Sistemas Distribuidos

- ✓ Recuperación de Fallos

- ✓ Tolerancia a Fallos

→ Seguridad en Sistemas Distribuidos

- ✓ Tipos de Amenazas y Ataques

- ✓ Políticas y Mecanismos

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Transacciones Distribuidas:**

- Conjunto de operaciones ejecutadas en sistemas distribuidos que garantizan la condición **ACID** ( *ACAP* ):

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Transacciones Distribuidas:**

- Conjunto de operaciones ejecutadas en sistemas distribuidos que garantizan la condición **ACID** ( *ACAP* ):

**Atomicity** ( *Atomicidad* ):

la transacción se realiza completa o no se realiza nada.

**Consistency** ( *Consistencia* ):

los estados anterior y posterior a la transacción son estados estables.

**Isolation** ( *Aislamiento* ):

los estados intermedios de la transacción son sólo visibles dentro de la propia transacción.

**Durability** ( *Permanencia* ):

las modificaciones realizadas por una transacción completada son permanentes.

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Transacciones Distribuidas:**

- **Tipos:**

- ❑ **Transacciones de 1 Fase**

( *One-Phase Commit* )

- ❑ **Transacciones de 2 Fases**

( *Two-Phase Commit* o *2PC* )



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Transacciones Distribuidas:**

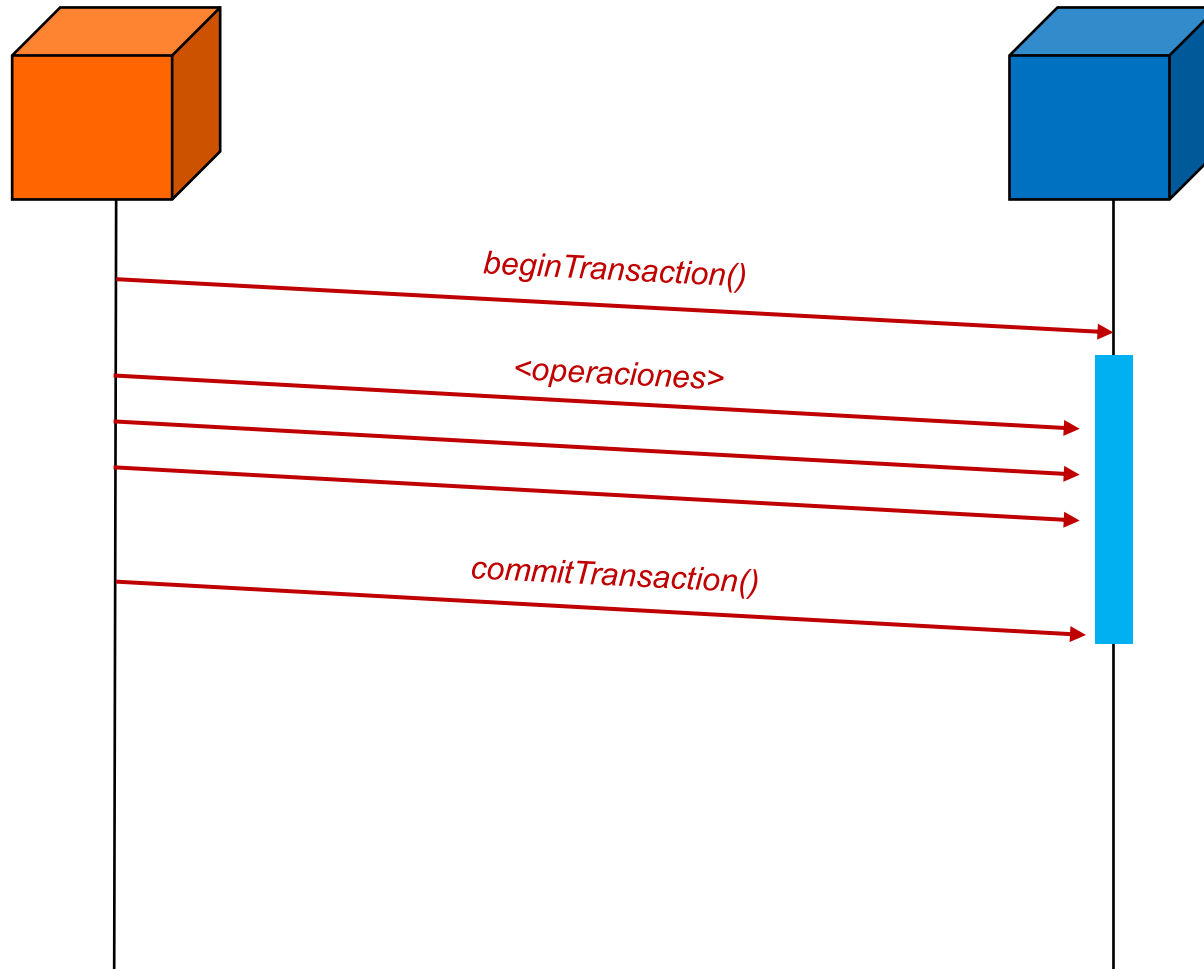
### ❑ Transacciones de 1 Fase:

- Aplica tres funciones especiales:
  - **beginTransaction()** : comienza un bloque de operaciones que corresponden a una transacción.
  - **commitTransaction()** : concluye el bloque de operaciones que conforman la transacción. Todas las operaciones se completan.
  - **abortTransaction()** : en cualquier punto se aborta la transacción y se regresa al estado anterior al comienzo de transacción.

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Transacciones Distribuidas:

### ❑ Transacciones de 1 Fase:



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Transacciones Distribuidas:**

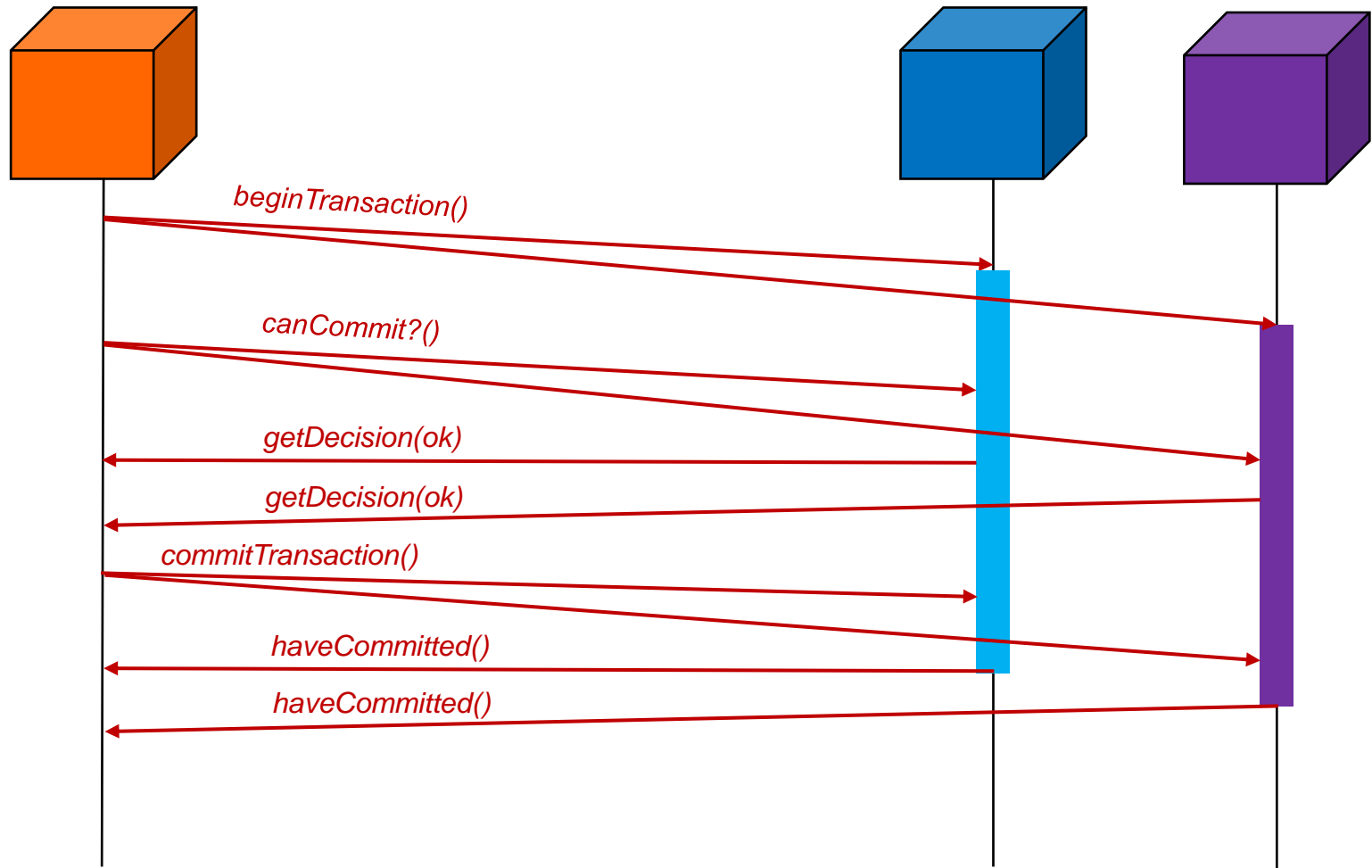
### ❑ Transacciones de 2 Fases:

- Soporta la posibilidad de ejecutar la transacción en más de un nodo en forma concurrente ( se hace en todos o en ninguno).
- El nodo que ejecuta la transacción actúa como '*Coordinador*'.
- Agrega las siguientes funciones especiales:
  - **canCommit?()** : el coordinador consulta si se puede completar la transacción.
  - **getDecision(rta)** : se notifica que la operación se puede completar o no.
  - **haveCommitted()** : se notifica que la operación se ha completado con éxito.

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Transacciones Distribuidas:

### ❑ Transacciones de 2 Fases:



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Manejo de Fallos en Sistemas Distribuidos:

- FALLO:

- Estado o situación en la que se encuentra un elemento cuando deja de cumplir la función para el cual había sido diseñado.
- Clasificación:
  - *Fallo del Sistema*
  - *Fallo de Almacenamiento*
  - *Fallo de Comunicación*
  - *Fallo de Proceso*

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Manejo de Fallos en Sistemas Distribuidos:

- Recuperación de Fallos:
  - Capacidad del sistema para volver a un *estado normal* ante un *fallo*.
  - Estrategias:
    - *Recuperación hacia Adelante*
    - *Recuperación hacia Atrás*
      - *Basado en Operaciones Transaccionales*
      - *Basado en Estados*
      - *Mixto*

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Manejo de Fallos en Sistemas Distribuidos:

- Tolerancia a Fallos:
  - Capacidad del sistema para evitar (o reducir) el impacto de un *fallo*.
  - Estrategias:
    - *Enmascaramiento de Fallos*
    - *Comportamiento bien definido ante Fallos*
    - *Resiliencia de Procesos ( 'Process Resilience' )*

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ **Seguridad en Sistemas Distribuidos:**

### □ Tipos de Amenazas y Ataques:

- ❖ Interrupción

- ❖ Intercepción

- ❖ Modificación & Fabricación



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:



### ❑ Políticas de Seguridad:

Establecen límites definidos en el manejo de los recursos y son independientes de la tecnología aplicada.

- Confidencialidad
- Integridad

### ❑ Mecanismos de Seguridad:

Determina la forma en que se implementan las políticas de seguridad. Se encuentran formados por conjunto de técnicas dependientes de la tecnología.

- Autenticación 
- Encriptación de Mensajes 
- Matriz de Acceso
- Firewalls

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

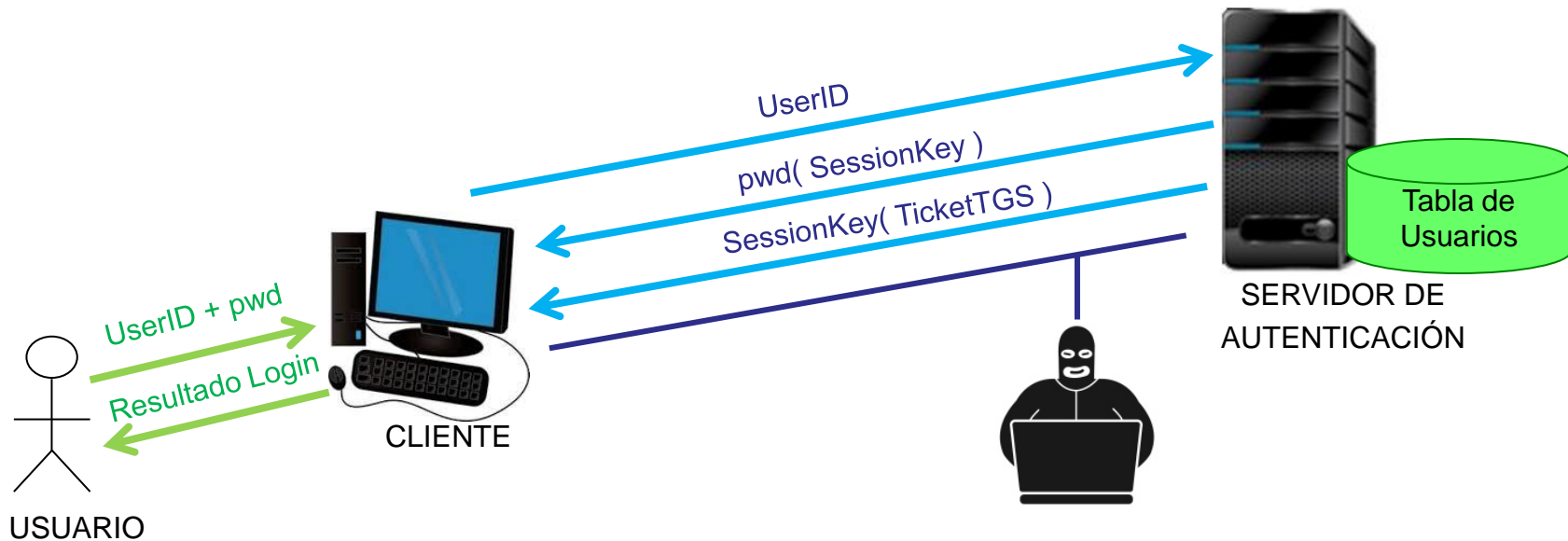
### ▪ Autenticación en Redes de Computadoras: **Kerberos**

- Mecanismos de autenticación desarrollado por el MIT en los años 80.
- Actualmente su uso es muy extendido (DCE, NFS, AFS-3, Windows, FreeBSD, MacOS, RedHat Linux, Solaris, AIX, Z/OS, HP-UX , OpenVMS ).
- Existe versión de código fuente disponible [<http://web.mit.edu/kerberos/>].
- Está basado en dos servicios:
  - *Servicio de Autenticación (AS)*  
Autentica a los usuarios en el login y expide tickets para el acceso al TGS
  - *Servicio de Expedición de Tickets (TGS)*  
Expide tickets y claves de sesión para el acceso de los usuarios a servicios específicos.

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

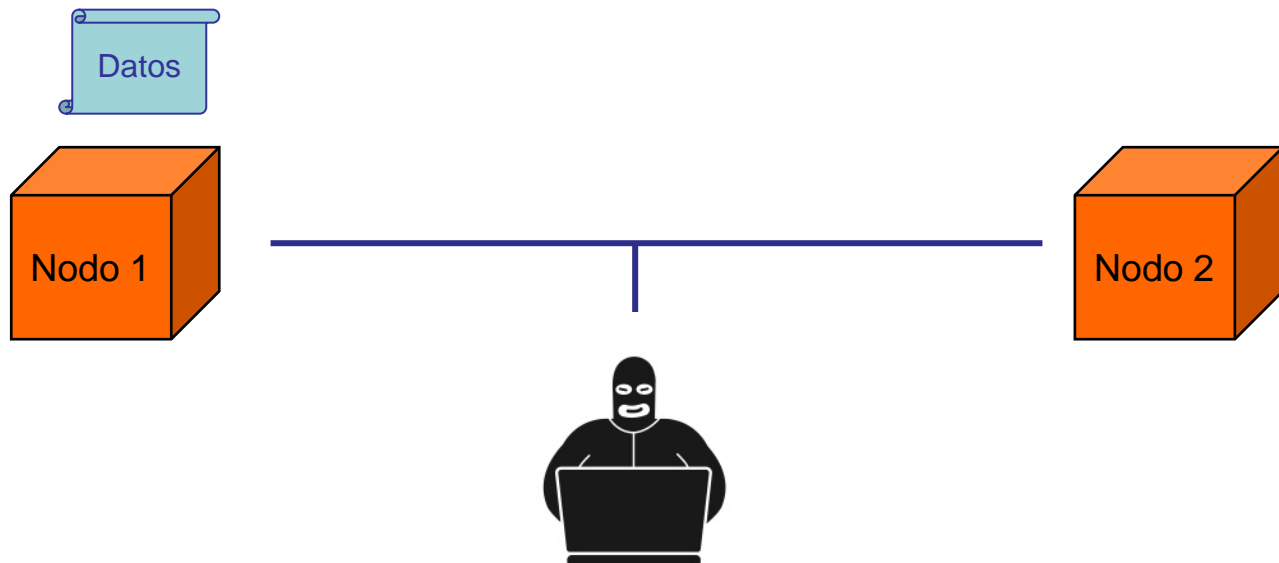
- Autenticación en Redes de Computadoras: **Kerberos**
  - Autenticación del Usuario:



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

- Encriptación de Mensajes:
  - Busca intercambiar mensajes entre 2 o más nodos asegurando la confidencialidad e integridad de la información.

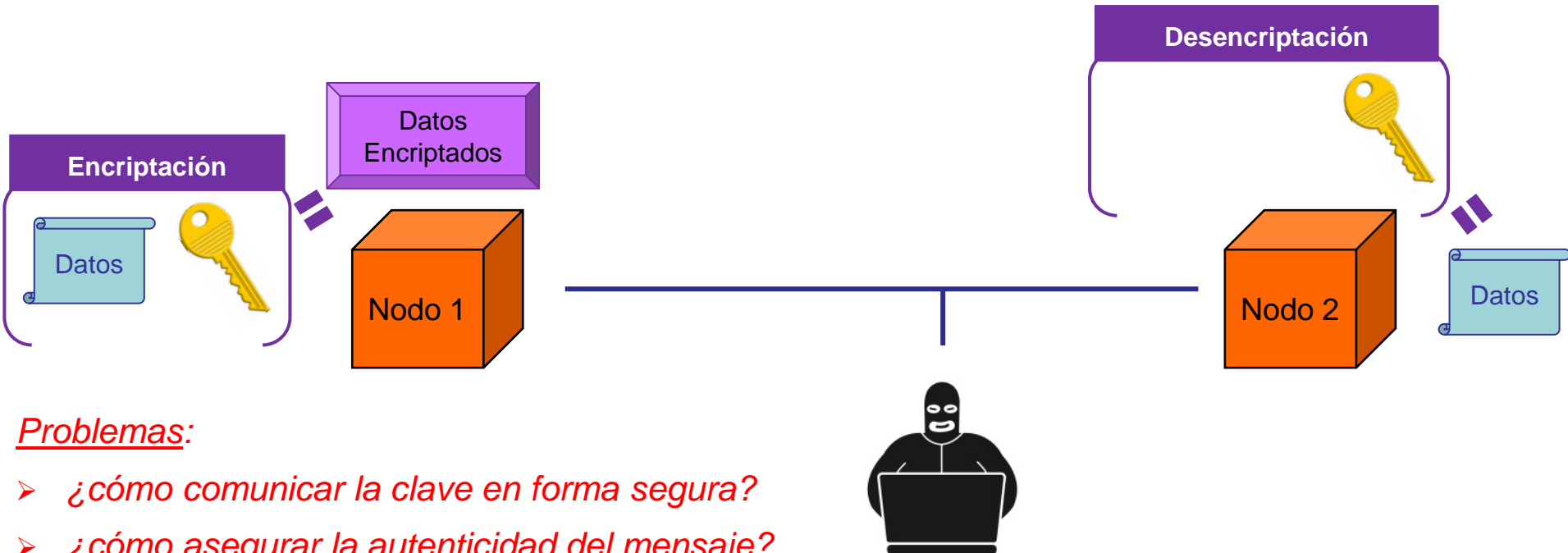


# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

### ▪ Encriptación de Mensajes:

- Busca intercambiar mensajes entre 2 o más nodos asegurando la confidencialidad e integridad de la información.
- Se puede aplica un algoritmo y una clave conocido por ambos nodos.



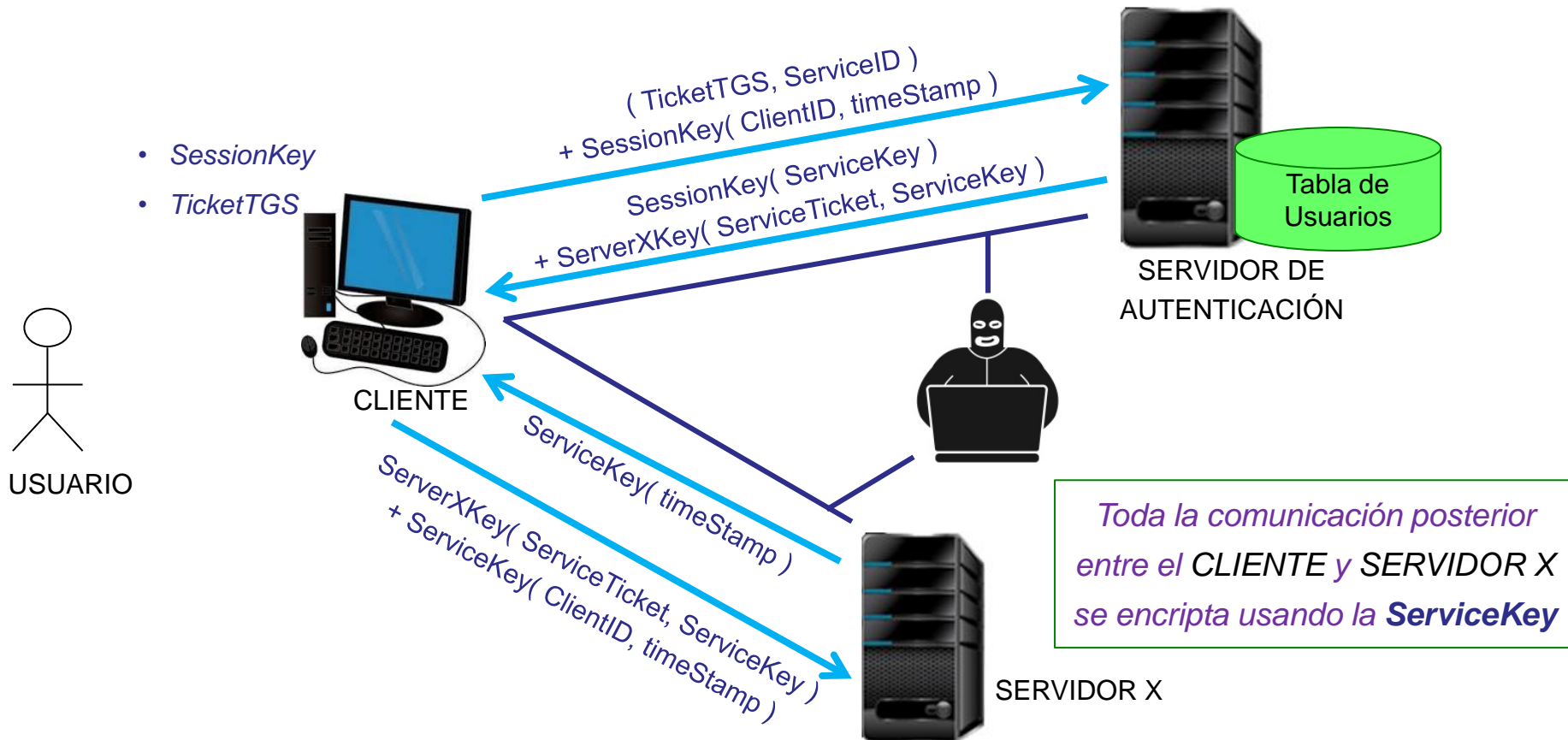
### Problemas:

- ¿cómo comunicar la clave en forma segura?
- ¿cómo asegurar la autenticidad del mensaje?

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

- Encriptación de Mensajes: **Kerberos**
  - Acceso a Servicios:



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

### ▪ Encriptación de Mensajes: **Digital Envelope**

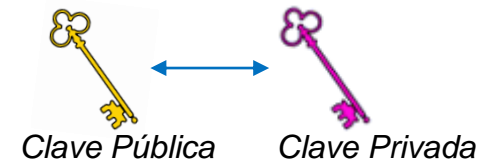
- Mecanismo aplicado para llevar a cabo el intercambio de mensajes y claves por un canal poco seguro.

- Se basa en la utilización de

- Método de Encriptación Simétrica



- Método de Encriptación Asimétrica (RSA)

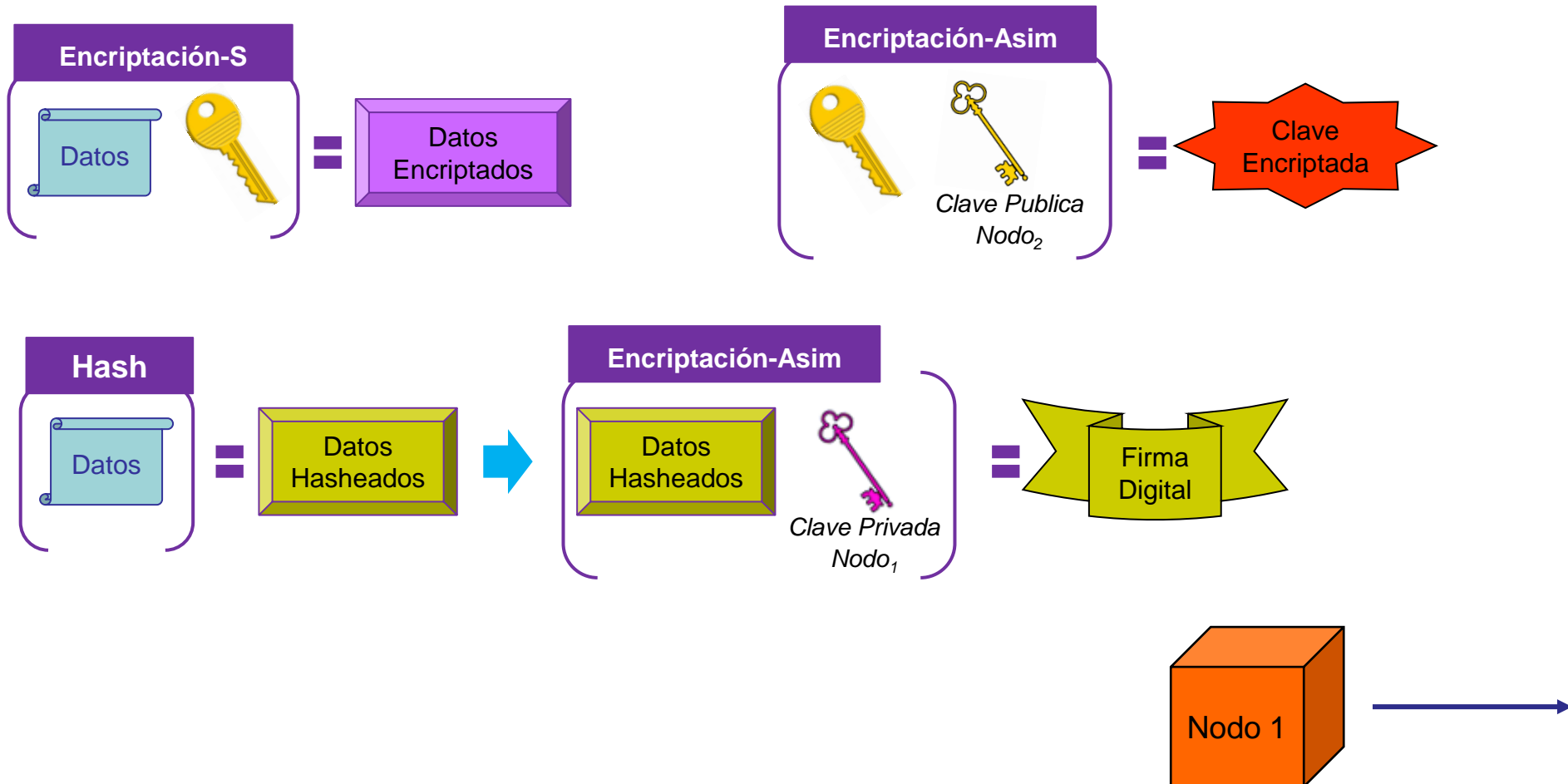


- Algoritmo Hash

# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

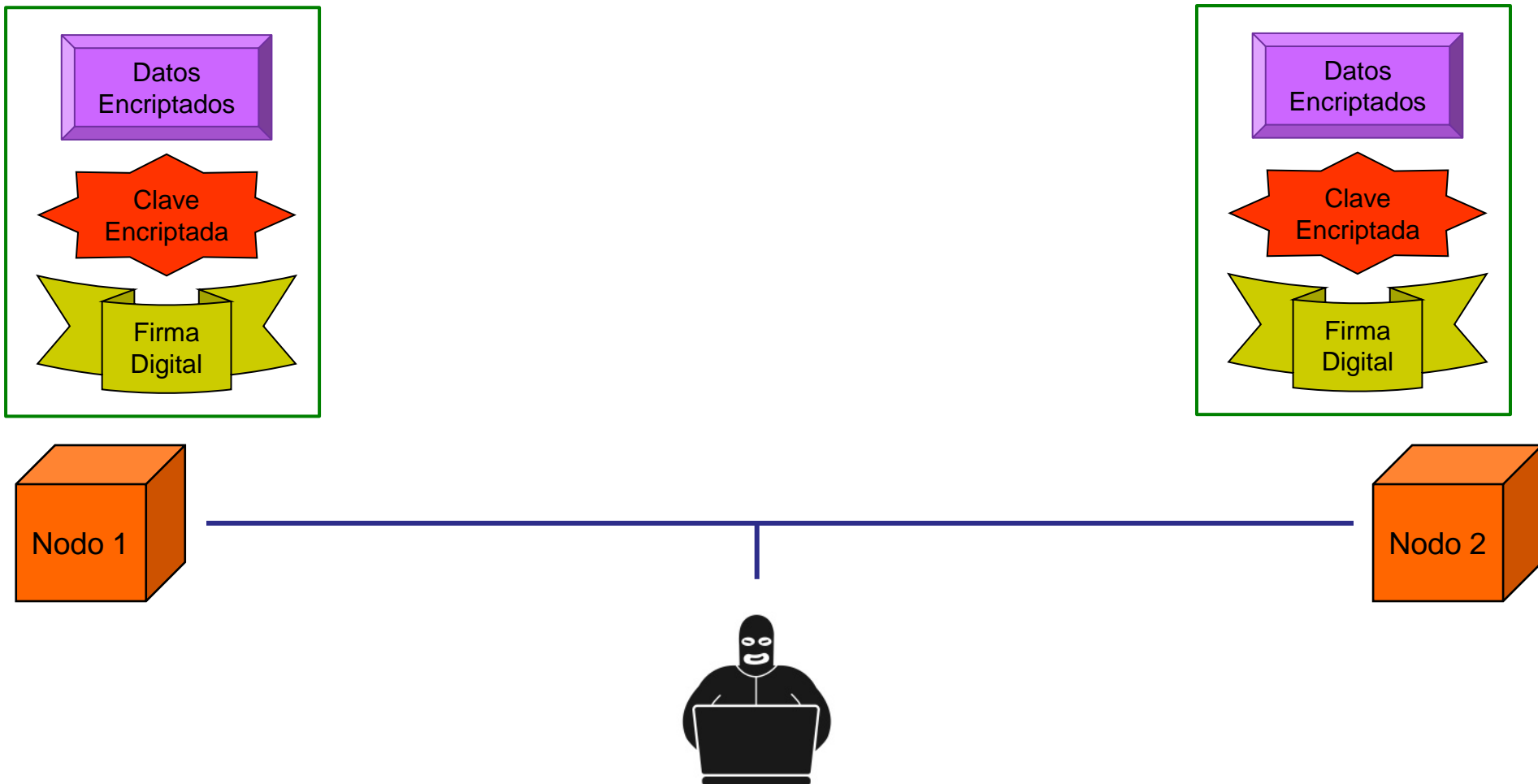
- Encriptación de Mensajes: **Digital Envelope**





# SISTEMAS OPERATIVOS DISTRIBUIDOS

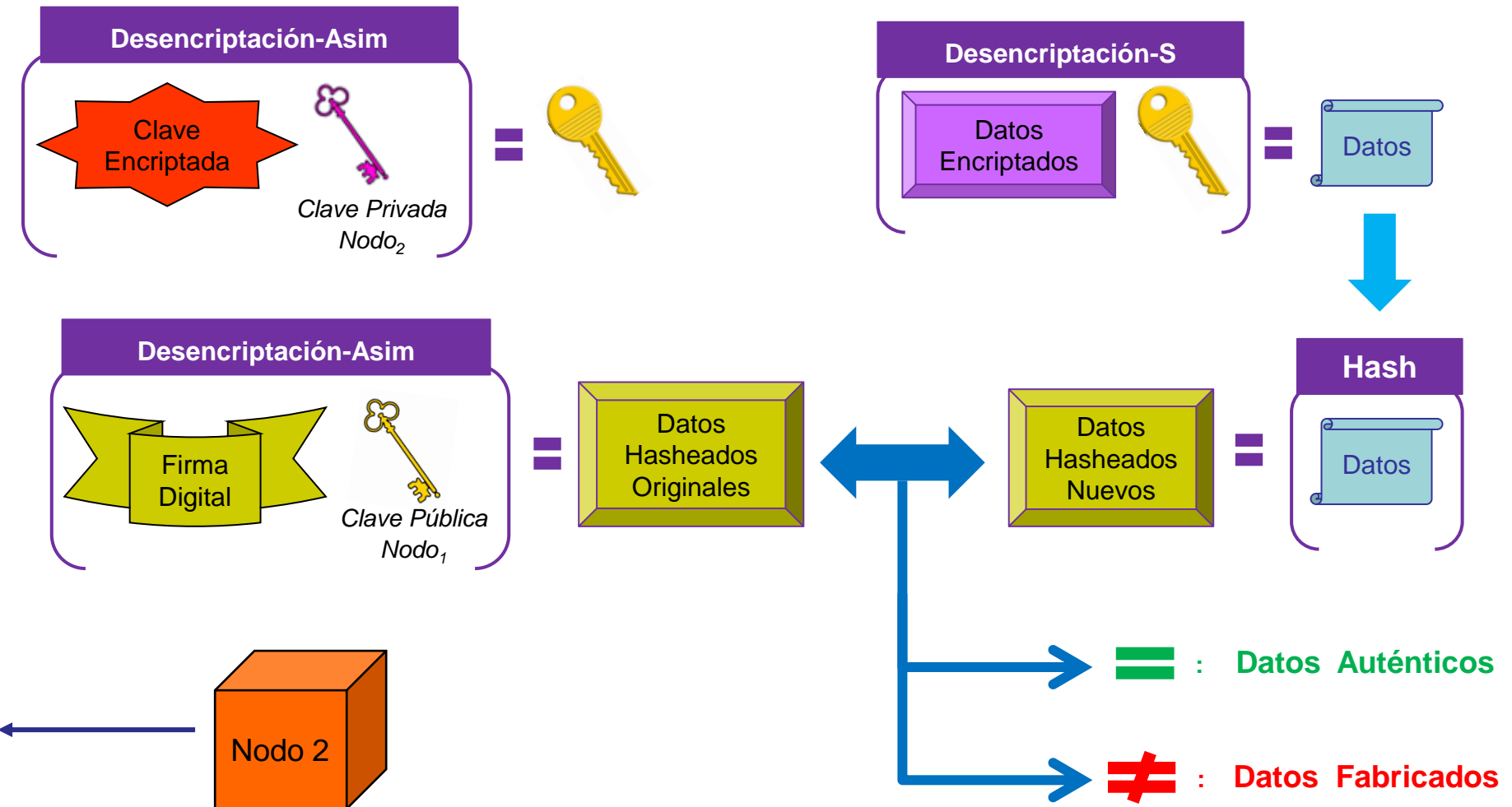
- **Seguridad en Sistemas Distribuidos:**
  - Encriptación de Mensajes: **Digital Envelope**



# SISTEMAS OPERATIVOS DISTRIBUIDOS

## ➤ Seguridad en Sistemas Distribuidos:

- Encriptación de Mensajes: **Digital Envelope**



- Guía de Estudio N° 6: *Seguridad y Transacciones en Sistemas Operativos Distribuidos* <http://sistemas.unla.edu.ar/sistemas/sls/ls-4-sistemas-operativos/pdf/SO-GE6-Seguridad-en-SOD.pdf>
- Singhal, M., & Shivaratri, N. G. (1994). Advanced concepts in Operating Systems. McGraw-Hill, Inc.. Capítulos 12 y 13.
- Stallings, W. (2005). Sistemas Operativos - Aspectos Internos y Principios de Diseño, 5<sup>ta</sup> Edición Prentice Hall. Capítulo 14.
- Tanenbaum, A.S. (2009). Sistemas Operativos Modernos, 3<sup>ra</sup> Edición Prentice Hall. Capítulo 13 (sección 13.6.5).

# Preguntas



**¡¡GRACIAS!!**

