* Cuestiones para implementar un sistema distribuido:
* como asegurar la confiabilidad y fiabilidad?
* Transacciones distribuidas

Conjunto de operaciones ejecutadas en sistemas distribuidos que garantizan la condición ACID:

Atomicidad: la transacción se realiza completa o no se realiza nada

Consistencia: los estados anterior y posterior a la transacción son estados estables

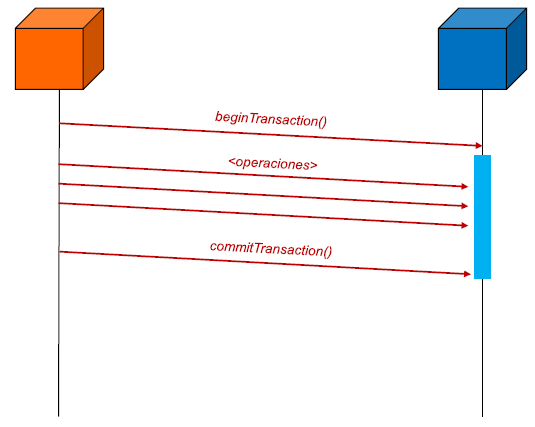
Aislamiento: los estados intermedios de la transacción son solo visibles dentro de la propia transacción

Permanencia: las modificaciones realizadas por una transacción completada son permanentes

* Tipos:
* Transacciones de 1 fase:

Aplica tres funciones especiales:

* + Begintransaction (): comienza un bloque de operaciones que corresponden a una transacción
  + Commitransaction (): 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



* 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).

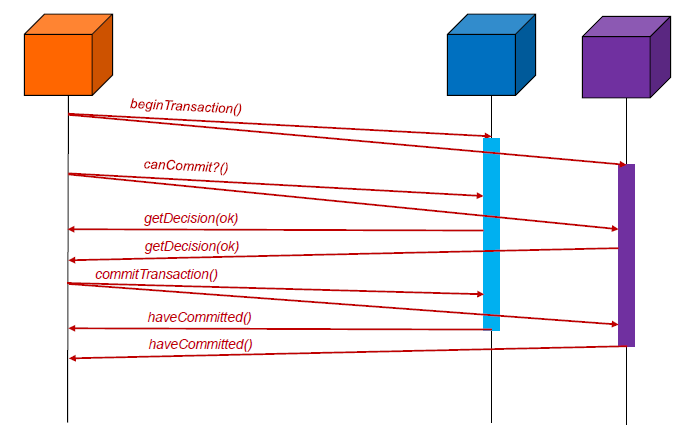
\*El nodo que ejecuta la transacción actúa como ‘*Coordinador*’.

\*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.



* 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 (falla el so o todo el sist distribuido)*

‒*Fallo de Almacenamiento (no podemos acceder info almacenada o actualizar los datos)*

‒*Fallo de Comunicación (un nodo no puede comunicarse a través de mensajes con otro nodo)*

‒*Fallo de Proceso (un proceso ejecutando dentro del sist distribuido no puede funcionar por x motivo)*

* Recuperación de Fallos:

• Capacidad del sistema para volver a un *estado normal* ante un *fallo*.

* Estrategias:

O *Recuperación hacia Adelante (mediante operaciones el so aplica alguna accion para corregir el fallo)*

O *Recuperación hacia Atrás (un rollback del estado del sistema, vuelve a una situación donde el fallo no se produjo)*

‒*Basado en Operaciones Transaccionales*

‒*Basado en Estados*

‒*Mixto*

* Tolerancia a Fallos:

•Capacidad del sistema para evitar (o reducir) el impacto de un *fallo*.

•Estrategias:

O *Enmascaramiento de Fallos (de acuerdo a los distintos fallos que se producen, va definiendo operaciones que se puedan realizar, logrando que los procesos sigan trabajando)*

O *Comportamiento bien definido ante Fallos (tanto el so como los procesos, tienen que indicar que deben hacer cuando se produce un fallo, es responsabilidad de los procesos)*

O *Resiliencia de Procesos* (*Process Resilience*) (los procesos ya sepan cómo deberían funcionar ante varios fallos, y así evitar completamente una tolerancia con error, es responsable el desarrollador de cómo se debe proceder ante un fallo).

* Seguridad en sistemas distribuidos:
  + - * Tipos de Amenazas y Ataques:
        + Interrupción (si tengo 2 o más nodos conectados en una red, esos nodos no pueden mandarse mensajes entre sí, porque hay un ataque por un ente externo de la red de forma fisica o lógica.)
        + Intercepción (el ente externo, monitorea y accede a los mensajes que se mandan los distintos nodos del sistema distribuido, así leerá dicha información sin permisos)
        + Modificación & Fabricación (el ente externo, modifica los mensajes de los nodos o fabrica nuevos mensajes que el nodo originario nunca envió)
    - Políticas de Seguridad:

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

* Confidencialidad (personas autorizadas o procesos, puedan acceder a cierta información)
* Integridad (personas autorizadas o procesos, puedan modificar o borrar información)
* Mecanismos de Seguridad:

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

* Autenticación en redes de computadoras

**Kerberos**

• Mecanismos de autenticación desarrollado por el MIT en los años 80.

• Actualmente su uso es muy extendido (Windows, FreeBSD, MacOS, RedHat Linux, Solaris).

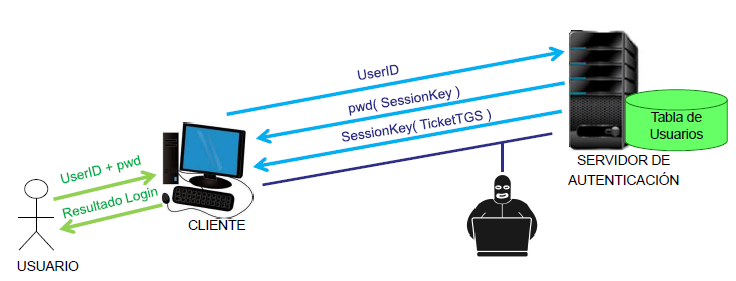
•Existe versión de código fuente disponible [http://web.mit.edu/kerberos/].

•Está basado en dos servicios:

‒*Servicio de Autenticación* (*AS*) Autentica a los usuarios en el login y expide tickets para el acceso al TGS

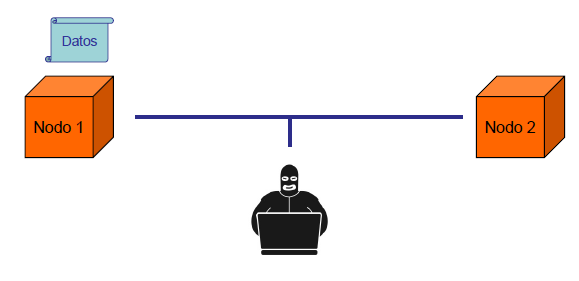
‒*Servicio de Expedición de Tickets* (*TGS*) Expide tickets y claves de sesión para el acceso de los usuarios a servicios específicos.

* Autenticación del Usuario:

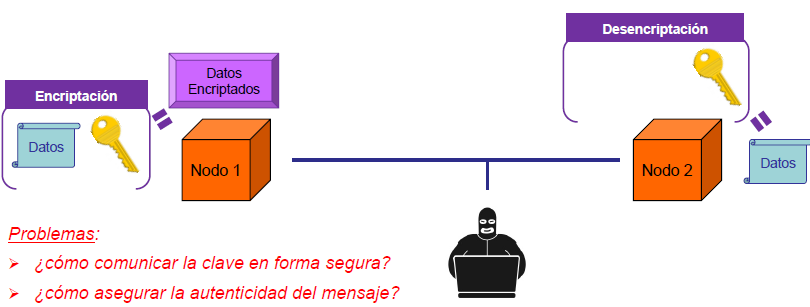


* Encriptación de Mensajes

•Busca intercambiar mensajes entre 2 o más nodos asegurando la confidencialidad e integridad de la información.



•Se puede aplicar un algoritmo y una clave conocido por ambos nodos.

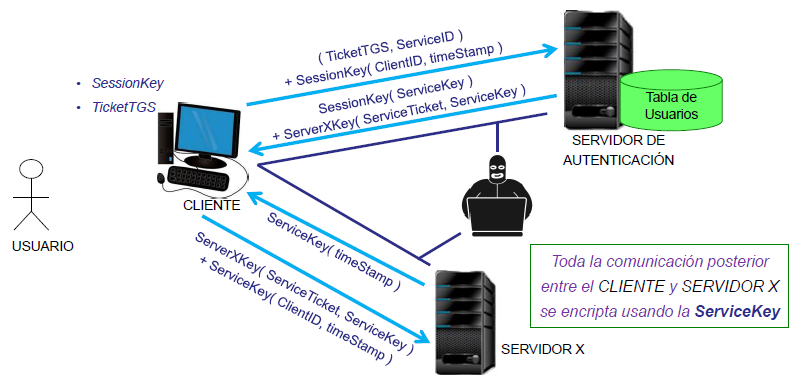


Para solucionar dichos problemas mencionados en la imagen, se utiliza 2 métodos:

**Kerberos**

* Acceso a Servicios:

**Digital Envelope**



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

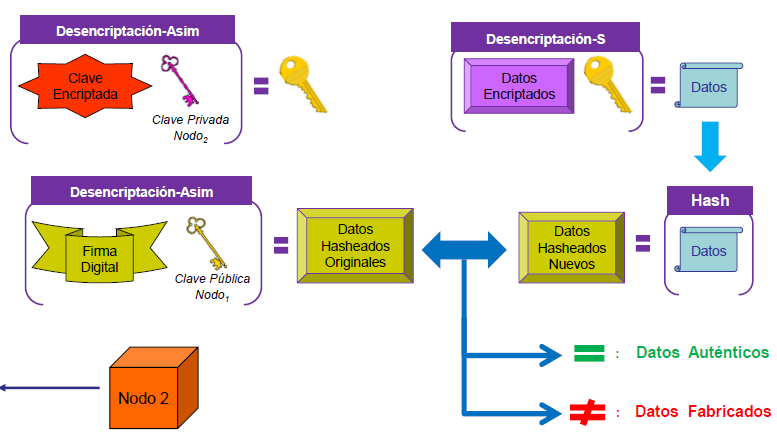
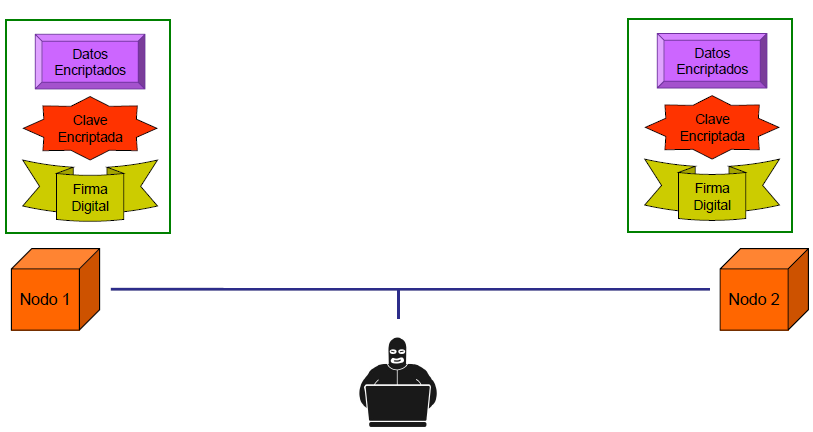
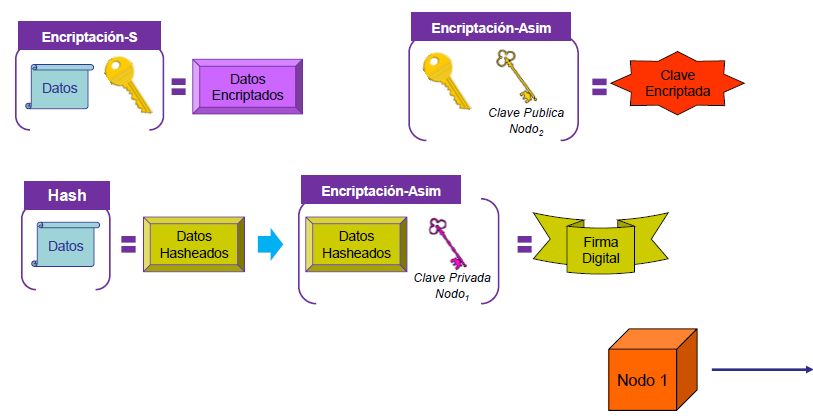
•Se basa en la utilización de

‒Método de Encriptación Simétrica (se utiliza la misma contraseña para desencriptar o encriptar, conocido como clave simétrica)

‒Método de Encriptación Asimétrica (RSA) (tengo únicamente una clave pública, la conoce todo el mundo asociado al usuario, y una clave privada, solo el usuario conoce; si encripto algo con una clave pública debo desencriptarlo con una clave privada, caso contrario viceversa, ambas claves están asociadas)

‒Algoritmo Hash (codificar una serie de datos en un conjunto de datos más pequeños; podría confirmar que un dato llego de forma correcta desde un usuario o desde un servidor)

Explicado estos 3 métodos, veremos cómo se implementan mediante el digital Envelope a través de estas tres imágenes:



* Matriz de Acceso
* Firewalls