ESTRATEGIAS DE MONITOREO DE BASE DE DATOS

Marko Antonio Rivas Rios

Jorge Luis Mamani Maquera

Orlando Antonio Acosta Ortiz

Yofer Nain Catari Cabrera

Orestes Ramirez Ticona

Roberto Zegarra Reyes

Universidad Privada de Tacna

Julio 07, 2019

Abstract

La teoría de la planificación del desarrollo define el seguimiento o monitoreo como un ejercicio destinado a identificar de manera sistemática la calidad del desempeño de un sistema, subsistema o proceso a efecto de introducir los ajustes o cambios pertinentes y oportunos para el logro de sus resultados y efectos en el entorno. Identificar los éxitos o fracasos reales o potenciales lo antes posible y hacer ajustes oportunos a la ejecución. Con un extendido consenso sobre la finalidad del monitoreo, como se define en el párrafo anterior, en la actualidad existen dos tendencias sobre el significado y el alcance de los sistemas de seguimiento o monitoreo. Una tendencia enfatiza la coincidencia entre lo planificado y lo ocurrido.

I. Introducción

La tecnología llego para complementar y completar la virtualización de servidores es la virtualización de contenedores de aplicaciones. Esta tecnología va un paso mas allá en el paradigma de la virtualización, permitiendo no sólo el salto de virtualizar servidores sino también de virtualizar directamente un contenedor donde se ejecuta una aplicación, permitiendo de este modo una mayor abstracción aislando la componente "lógica de la aplicación" del componente “sistema operativo”.

II. Objetivos

Se busca saber un poco mas sobre:

AUDITORIA DE BASES DE DATOS:

Es el proceso que permite medir, asegurar, demostrar, monitorear y registrar los accesos a la información almacenada en las bases de datos incluyendo la capacidad de determinar: • Quién accede a los datos
• Cuándo se accedió a los datos.
• Desde qué tipo de dispositivo/aplicación.
• Desde qué ubicación en la red.
• Cuál fue la sentencia SQL ejecutada.
• Cuál fue el efecto del acceso a la base de datos.

Es uno de los procesos fundamentales para apoyar la responsabilidad delegada a TI por la organización frente a las regu-

laciones y su entorno de negocios o actividad.

TIPOS DE AUDITORÍAS DE BASES DE DATOS

Consideramos dos grandes tipos de auditorías de la base de datos, esta división está relacionada directamente con las actividades que los usuarios realizan, suponga que un usuario desea cambiar la dirección de envío de pedidos a un cliente. Para lograr esto se requiere que el usuario realice una serie de pasos, por ejemplo:

P 1. Conectarse a la base de datos.
P 2. Ejecutar el cambio de dirección.
P 3. Desconectarse.

Los pasos 1 y 3 se lograrían si cuenta con los privilegios o derechos necesarios para conectarse o desconectarse; para el paso 2 el usuario debe tener privilegios de acceso a la tabla de clientes, y la posibilidad de hacer una modificación. Basándonos en esta secuencia de pasos establecemos el primer tipo de auditoría, y la llamamos auditoría de actividades de los usuarios.

El segundo tipo de auditoría está inmersa en el paso 2. ¿Qué ocurrió realmente cuando cambió el dato, qué valores quedaron?, ¿qué valores había?, en otras palabras, ¿qué cambios produjo la transacción? Los controles que establezcamos para conocer lo que realmente ocurrió los llamaremos auditoría de transacciones.

Auditoría de actividades

El primer tipo de auditoría lo llamamos au ditoría de actividades, que consiste en controlar las actividades que realizan los usuarios en los objetos de la base de datos y entenderemos como objetos todas las tablas, vistas, restricciones de integridad que los usuarios crean en la base de datos.

Este proceso de monitoreo de las actividades de los usuarios permite encontrar posibles accesos a objetos no autorizados, conexiones en horas o días fuera de horarios normales. Toda esta actividad se va almacenando en una tabla o en un archivo que llamaremos el registro de auditoría.

El registro de auditoría (RA) tiene un crecimiento muy alto, por lo que es necesario administrarlo adecuadamente, muchas veces este registro llega a ser igual o mayor en tamaño que la base de datos. Imagínese el RA de una organización de más de 2000 empleados, que además permite el acceso de clientes por Internet, o el registro de auditoría de un banco, que puede recibir más de 20 transacciones por minuto.

Auditoría de transacciones

La auditoría de transacciones consiste en implementar una serie de controles que permiten llevar una bitácora de todas las transacciones que los usuarios realizan, pero a un nivel tal que podamos establecer una historia de cómo se produjeron los cambios. Al igual que en el tipo anterior, es necesario crear un registro de auditoría al que llamaremos registro de auditoría de transacciones (RAT). El crecimiento del RAT es superior al del RA, ya que es posible que un usuario que se conecte una sola vez, pueda hacer 30 transacciones. En este caso el RA almacena las actividades de conexión y desconexión, básicamente mientras que el RAT almacena 30 registros correspondientes a la información antes y después de cada transacción.

Sistemas de Gestión de Bases de Datos (SGBD).

Entre los componentes del SGBD podemos destacar el núcleo (kernel), el catálogo (componente fundamental para asegurar la seguridad de la base de datos), las utilidades para el administrador de la base de datos (entre la que se pueden encontrar algunas para crear usuario, conceder priv-

ilegios y resolver otras cuestiones relativas a la confidencialidad); las que se encargan de la recuperación de la BD: rearranque, copias de respaldo, ficheros diarios (log), etc. Y algunas funciones de auditoría, así como los lenguajes de la cuarta generación (L4G) que incorpora el propio SGBD.

En cuanto a las funciones de auditoría que ofrece el propio sistema, prácticamente todos los productos del mercado permiten registrar ciertas operaciones realizadas sobre la base de datos de un fichero (o en un conjunto de tablas) de pistas de auditoría (audit trail). El propio modelo de referencia de gestión de datos -ISO (1993)- considera las pistas de auditoría como un elemento esencial de un SGBD, señalando que —el requisito para la auditoría es que la causa y el efecto de todos los cambios de la base de datos sean verificables". El auditor deberá revisar, por lo tanto, la utilización de todas las herramientas que ofrece el propio SGBD y las políticas y procedimientos que sobre su utilización haya definido el administrador, para valorar si son suficientes o si deben ser mejorados.

Software de Auditoría.
Son paquetes que pueden emplearse para facilitar la labor del auditor, en cuanto a la extracción de datos de la base, el seguimiento de las transacciones, datos de prueba, etc. Hay también productos muy interesantes que permiten cuadrar datos de diferentes entornos permitiendo realizar una verdadera —auditoría del dato.

Sistema de Monitorización y Ajuste (Tunning)
Este tipo de sistema complementan las facilidades ofrecidas por el propio SGBD, ofreciendo mayor información para optimizar el sistema, llegando a ser en determinadas ocasiones verdaderos sistemas expertos que proporcionan la estructura óptima de la base de datos y de ciertos parámetros del SGBD y del SO. La optimización de la base de datos, como ya hemos señalado, es fundamental, puesto que se actúa en un entorno concurrente puede degradarse fácilmente el nivel de servicio que haya podido establecerse con los usuarios.

Características principales de un sistema de auditoría de base de datos

El sistema de auditoría de base de datos debe cumplir con ciertas características como las siguientes:
SISTEMA CONFIABLE E INTEGRAL

Se deben poder garantizar esquemas de auditoría contínua todos los 365 días del año, es decir:

• El volumen de la información relacionada con las trazas de auditoría puede ser muchas veces más grande que el tamaño de la base de datos a auditar.
• Se debe poder controlar el acceso y la modificación a las trazas almacenadas.
• Se debe poder proteger la información almacenada en las trazas de auditoría

Se deben poder auditar el acceso a la base de datos desde todas las posibles capas de acceso:
Aplicación / Front End:
La capa de aplicación es utilizada por usuarios calificados y posiblemente por usuarios maliciosos. Se pueden explotar las debilidades de las aplicaciones. Por ejemplo con la inyección SQL.

Servidor de Aplicación / Web Server:
Los usuarios con privilegios pueden tener acceso directo utilizando funciones del servidor de aplicaciones. Usuarios maliciosos pueden aprovechar las debilidades de los servidores de aplicaciones. Pueden

existir caballos de Troya listos para actuar en el código de las aplicaciones.
Manejador de BDD:
Pueden realizarse accesos no autorizados ODBC a través de la Red; Accesos no autorizados de usuarios con privilegios. Vulnerabilidades conocidas de los manejadores de base de datos.
Sistema Operativo:
Tenemos los accesos directos a los archivos en el sistema operativo. **CAPAZ DE CONSOLIDAR LAS TRAZAS DE AUDITORÍA**

Las trazas de auditoría deben tener:
• Quién realizó la operación
• Desde donde se realizó la operación (Dirección IP / Host o Aplicación)
• Cuándo se realizó la operación
• Qué se hizo durante la operación (información antes y después)
• Por qué se hizo la operación (contexto sobre el que se realiza la operación)

REGLAS DE AUDITORÍA BASADAS EN NECESIDADES ESPECÍFICAS

Las trazas de auditoría pueden llegar a tener tamaños inmanejables (cientos de terabytes de información).
Cada proceso de auditoría requiere su propia definición de reglas de auditoría se deben utilizar:

• Qué método de auditoría
• Qué Servidor /manejador (Base de Datos / tablas)
• Qué usuarios (Objetos, Horas, Acciones)
• Qué acciones tomar
Tipos de eventos a auditar:

• Eventos tipo DDL (Data Definition Lenguaje): asociados con la creación de usuarios, roles, tablas, etc.
• Eventos tipo DML (Data Manipulation Lenguaje): insert, update, delete.
• Eventos tipo Select: consultas de información
• Recompilaciones de Scripts
• Auditoría de los usuarios con privilegios de acceso.

. **NO SE DEBE AFECTAR EL DESEMPEÑO DE LA BASE DE DATOS.**

. Una de las principales preocupaciones relacionada con la auditoría de base de datos es su impacto en el desempeño de las aplicaciones. Es decir:
.
. • Impacto en el tiempo de respuesta de las aplicaciones
. • Impacto sobre la utilización de los manejadores de base de datos
. • Impacto sobre el espacio en los servidores
. **CAPAZ DE GENERAR NOTIFICACIONES EN TIEMPO REAL**
.
. Para lograr esquemas más efectivos de auditoría es necesario poder generar notificaciones en tiempo real. Mediante Email o Sistemas de monitoreo central.

. **CAPACIDAD PARA RETENER TRAZAS POR LARGOS PERÍODOS DE TIEMPO**
.
. Como el número de trazas puede ser muy grande se necesitan esquemas de archivos y preservación, como son:
. • Compresión de la información
. • Movimiento a cintas
. • Rutinas de eliminación
.
ADMINISTRABLE Y ESCALABLE EN EL TIEMPO
.
Debe realizarse una planificación de la Auditoría de base de datos que permita:

1. Identificar la base de datos de la institución
2. Clasificar los niveles de riesgo de los datos en la base de datos

3. Analizar los permisos de acceso
4. Analizar los controles existentes de acceso a la base de datos
5. Establecer los modelos de auditoría de BD a utilizar
6. Establecer las pruebas a realizar para la BD, aplicación y/o usuario
7. Alcance de la Auditoría (Selección de: tablas, usuarios, horario a auditar)
8. Tipos de Acciones (Solo registro de eventos, solo generación de alertas, Generación y registro de alertas)
9. Reportes a producir (Contenido y Frecuencia)

MONITOREO

La opción de Auditoría de Sistema tiene además la opción de Monitoreo, la cual permite visualizar la información de auditoría de sistema de BDD Oracle que fue configurada en las opciones anteriores. El monitoreo se realiza para: Sesiones y Objetos (acciones realizadas sobre estos).
Al escoger la opción de Sesiones del menú monitoreo de Sistema (fig. 22) obtenemos como resultado la siguiente información (Ver figuras: 23, 24, 25, 26) : Nombre de usuario, código del Terminal, Nombre del Usuario de Sistema Operativo, Estado de conexión, y un menú navegable que contiene información adicional como: Fecha y hora de conexión de usuario para Ingreso y salida creadas por AUDIT SESSION (ver fig. 23); Acción [contiene el nombre del tipo de acción : ACTION NAME] , Userhost [contiene el nombre de la máquina anfitrión del cliente]y Log off (ver fig. 24); Logoff Lread [es la lectura lógica de la sesión] , Logoff Pread [es la lectura física de la sesión] y Logoff Lwrite [es la escritura lógica de las sesión](Ver fig. 25); Logoff Dlock [son los puntos muertos detectados durante la sesión] y Sessionid [es el identificador numérico para cada sesión Oracle] (ver fig. 26). La información mostrada en las pantallas será la escogida durante la configuración de la auditoría de sistemas.

III. Conclusiones

Ambas tecnologías ofrecen ventajas distintas:
La virtualización viene con una plétora de herramientas probadas a lo largo del tiempo, plataformas de gestión y orquestación, sondas virtuales, soluciones de infraestructura virtual hiperconvertidas y mucho más. La portabilidad y la interoperabilidad son las características que destacan frente a los contenedores.

Los contenedores ofrecen una mayor eficiencia de recursos y agilidad de servicio. Aunque no parezca mucho, abre la puerta a un modelo de microservicios que puede escalar más rápido y de manera más eficiente. Los contenedores de papel se ajustan más a las iniciativas de NFV/SDN y la industria se ha dado cuenta de que Kubernetes es uno de los proyectos de código abierto de más rápido crecimiento hasta la fecha.

ESTRATEGIAS PARA LA SOLUCION DE PROBLEMAS

Aquí tenemos una lista de lo que debe hacer y en el orden que debe ser:
• Defina el problema
. se puede poner difícil ya que puede ser tan general como “todo está lento” o “el sistema no responde”. En el mejor caso es cuando el problema es específico y podemos tener algo como “Es un problema de bloqueo en esta tabla”. De todas maneras, esto debería ser el punto de partida. Yo no haría nada sin antes describir el problema lo más detallado posible
. • Analice si es interno o externo
. este es un paso muy importante porque puede ayudarnos a poder solucionar casi la mitad del problema. Especialmente si hay otras cosas ejecutándose en el SQL Server, eso puede abrir una olla de grillos completamente nueva. Así que, si es interna, estás a mitad del camino. Si no es así, bueno, entonces ya es otra historia,

pero de cualquier manera siempre pasaría por este paso
. • Determine si está vigente o está en curso
. si sabemos que es un problema interno (el problema es el servidor SQL), tenemos averiguar si este está vigente o en curso. Vigente significa que acaba de suceder, con esto estamos seguros de que esta es la primera vez que se produce un problema. Si está en curso, quiere decir que tal vez haya pasado una semana sin siquiera saber que este problema está allí. En este caso, los registros nos pueden ayudar.
. • Identifique y resuelva
. este punto es obvio y no necesita ninguna explicación. Con un poco de suerte y siguiendo los tres pasos anteriores, llegarás a solucionarlo rápidamente
.

References

1. http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/23/21
2. https://programarfacil.com/blog/que-es-un-orm/
3. https://www.beeva.com/beeva-view/tecnologia/mas-alla-de-la-virtualizacion-contenedores/
4. https://searchdatacenter.techtarget.com/es/definicion/virtualizacion-basada-en-contenedores-virtualizacion-a-nivel-de-sistema-operativo
5. https://www.incibe-cert.es/blog/asegurando-virtualizacion-tus-sistemas-control
6. http://www.datakeeper.es/?p=716