

# Asignatura: OPC13 – Cloud Computing

Ensayo de resultados de aprendizaje de la **semana 12**

Temas: Almacenamiento, serverless

*Integrantes:*

Diego Pérez Prieto  
Matrícula: 365341  
[a365341@uach.mx](mailto:a365341@uach.mx)

Jared Alejandro Rosas Molina  
Matrícula: 365337  
[a365337@uach.mx](mailto:a365337@uach.mx)

## **1. Ensayo Tema “Empezando con el Almacenamiento”**

Dentro de los almacenamientos de datos existen dos tipos, aquellos alojados en las instalaciones de la empresa, donde esta misma administra, controla y mantiene al servidor donde se alojan los datos, además estos últimos se comparten a través de la red local de la empresa, a su vez, también existe el almacenamiento en la nube, la cual es un tipo de almacenamiento que se contrata con un proveedor, éste último dará mantenimiento necesario al hardware y software que se utiliza para alojar los datos de la empresa.

El almacenamiento en la nube tiene distintos tipos de almacenamiento, como lo es por bloques, por archivos o por almacenamiento por objetos. El almacenamiento por bloques divide los datos por bloques de tamaño fijo, para luego, almacenarlos en piezas separadas con un identificador, permitiendo que se puedan guardar en distintos sistemas, en el almacenamiento por archivos, los datos se almacenan de manera jerárquica, la cual ayuda a los usuarios, las aplicaciones y los servicios a acceder a los datos compartidos, en el almacenamiento de objetos, los archivos se almacenan como objetos que se basan en atributos, donde cada objeto se compone de datos, metadatos (información acerca de los datos) y un identificador del objeto, cuando un archivo del objeto se cambia, se modifica todo el objeto. Este almacenamiento cuenta con distintas ventajas, como la paga por lo que se usa, es segura, se puede acceder desde distintos dispositivos, se puede aumentar o disminuir el tamaño utilizado además de que está respaldado, almacenando los datos en distintos servidores para su accesibilidad.

Un servicio que da almacenamiento en la nube es Amazon S3, el cual es un almacenamiento de objetos que sirve para almacenar, analizar y recolectar información. Dentro de este, los datos se almacenan como objetos en “buckets”, en los que los datos que pueden ser de distintos tipos (videos, imágenes, archivos de texto, entre otros), dándoles un identificador o clave con el cual se podrá utilizar el objeto en distintas aplicaciones, donde con la combinación la clave, el nombre del bucket y el id de la versión se permite una interacción con el archivo.

El almacenamiento por objetos se clasifica según la frecuencia con la que se acceden a los datos, los cuales son almacenamiento activo y de archivo, el activo se refiere a archivos que se utilizan todo el tiempo, pero también son aquellos que no se usan en gran frecuencia pero se necesita un acceso rápido a estos, el almacenamiento de archivo, son aquellos a los que se accede con poca frecuencia pero se tienen que mantener en el sistema, donde dentro del servicio de Amazon S3, existen distintos tipos de archivos que pueden variar según los datos que se contengan, por ejemplo, para los datos activos esta Amazon S3 Estándar, el cual se puede usar para los datos de acceso poco frecuente junto con One Zone, para almacenamiento de archivo esta Glacier y Glacier Deep Archive.

Dentro del servicio de Amazon S3, para la creación de un bucket debe de tener un nombre único globalmente, donde para acceder a este se hará a través de la combinación del nombre, la clave y el ID de la versión, luego se tendrá que escoger la región donde se localizará el bucket, donde estos pueden tener una mayor seguridad mediante listas de control de acceso (ACL) o bloqueando el acceso público, donde con las dos se administra quienes pueden ver y utilizar la información dentro de este, también los archivos pueden tener un control de versiones en dado caso de que sucedan errores por parte de la aplicación, desarrolladores o usuarios, permitiendo así una recuperación de los datos. Una vez creado el bucket, se podrán cargar los objetos dentro de este, de los cuales se podrá escoger su tipo de almacenamiento según su frecuencia, con los objetos se puede hacer una copia por distintos motivos, hacerle referencia mediante su URL, o inclusive descargarlos, a estos mismos se les puede añadir distintas reglas como ciclo de vida o replicarlos en distintos lugares. La seguridad de la información es responsabilidad tanto de AWS como del cliente, donde el cliente su obligación es la seguridad dentro de la página web, y en AWS, los buckets tienen distintos atributos que le permiten ser más seguros como Políticas de AWS Identity and Access Management, de buckets e inclusive cifrados.

Para el almacenamiento en bloques, se puede usar un almacenamiento temporal a nivel bloque en una instancia EC2, el cual ya está configurado y vinculado a un almacenamiento en disco en el servidor físico, o se puede utilizar Amazon EBS, el cual es un almacenamiento extraíble asociado con una zona de disponibilidad, donde si la instancia se acaba los datos seguirán existiendo, este servicio da distintas funciones y beneficios como es la disponibilidad, la persistencia, el cifrado y la seguridad de los datos, son flexibles, permitiendo cambios sin interrupciones y junto con esto, se pueden guardar datos de los volúmenes de EBS en S3 como respaldo, donde los volúmenes son unidades de estado sólido (SSD) o unidades de disco duro (HDD), donde los SSD se usan para transacciones con operaciones de escritura y lectura más frecuentes, mientras que las unidades de disco duro son utilizadas para cargas de trabajo de alto rendimiento. Para el almacenamiento en archivos, se puede utilizar el EFC, siendo este sistema escalable donde sus datos son permanentes, este sistema cuenta con distintos beneficios, donde está completamente administrado, es duradero y tiene alta disponibilidad, es elástico y escalable además de que los datos son cifrados, este principalmente es utilizado en administración de contenido, flujos de trabajo de procesamiento multimedia, análisis de big data, entre otros.

## **2. Ensayo Tema “Empezando con la computación sin servidores”**

Muchas compañías han dejado de construir sus aplicaciones como lo hacían en el pasado. Antes, dominaba la arquitectura monolítica (donde todo se construye como una sola unidad), pero ahora están emigrando a una arquitectura basada en microservicios. Este es un estilo de arquitectura donde la aplicación se divide en un conjunto de servicios pequeños e independientes. Cada uno se enfoca en una sola función (como "pagos", "usuarios" o "catálogo") y se comunican entre sí a través de APIs bien definidas. La ventaja de los microservicios es que cada componente o servicio puede ser ejecutada, actualizada y escalada de manera independiente para cumplir con las demandas de una aplicación en específico, si un componente falla, los demás componentes permanecen activos, y si necesitas actualizar algún componente, no necesitas actualizar los demás. En contraste, en las arquitecturas monolíticas si un componente falla, toda la aplicación se cae, y si se necesitara actualizar algún componente en específico, tenías que actualizar todos los demás componentes, y si se llega a caer algún componente, todos los demás se caen.

Aquí es donde entra la computación sin servidores. Con este modelo los desarrolladores se pueden enfocar en desarrollar y construir aplicaciones sin tener que preocuparse por gestionar los servidores, pudiendo enfocar su energía en construir mejores productos que sean escalables y confiables. Los serverless se pueden utilizar para construir cualquier tipo de aplicación o servicio backend y la nube será la encargada de gestionar toda la infraestructura de servidores. Además, los servicios sin servidores se pueden integrar correctamente con la mayoría de los servicios que sí usan servidores, por lo que se pueden complementar con aplicaciones tradicionales sin problema.

Un ejemplo mencionado en el curso es AWS Lambda, que es un servicio serverless que se puede usar para correr código para cualquier tipo de aplicación o servicio backend, sin tener que provisionar o administrar servidores. Con este servicio, los desarrolladores simplemente suben su código como una "función" y AWS se encarga automáticamente de encontrar un servidor, ejecutar el código cuando se necesita y "apagarlo" cuando termina.

La diferencia entre los microservicios y serverless, es que el primero es un método para construir la arquitectura, mientras que serverless es un servicio que es usado para construir la arquitectura. Los microservicios son el método por el cual se construye la arquitectura, y serverless son las herramientas que se usan para construir ese método.

La manera en la que se comunican los microservicios de forma eficiente es a través de la arquitectura orientada a eventos. En lugar de que el servicio A llame directamente al servicio B, el servicio A simplemente emite un evento, otros servicios están “escuchando” ese evento y reaccionan cada uno por su cuenta. Los serverless funcionan de esta manera, estos actúan cuando ocurre un evento o “trigger”. Este evento puede ser una visita a una API, la subida de un archivo o un evento emitido por otro microservicio.