Preguntas teóricas

1. ¿Cuál es la diferente entre nube pública, privada e híbrida?

   La nube pública es aquella en la que un proveedor o tercero brinda los servicios (infraestructura, almacenamiento y otros) mediante el internet y son dueños de la infraestructura en la cual está montado estos servicios la cual se comparte a los clientes. Adiciona a ello, el cliente puede hacer uso de estos servicios mediante una suscripción.
   La nube privada es aquella que sólo puede usarse por una empresa y se despliega sobre el datacenter de la empresa o alquilada a un proveedor y su consumo también se realiza por el internet.
   La nube híbrida es un mix entre la infra local y la pública las cuales se comunicar para compartir información o expandir la infra local.

2. Describa tres prácticas de seguridad en la nube

- contar con un plan de remediación de vulnerabilidades que se ejecute de forma períodica
- hacer uso efecto del IAM o RBAC para la asignación correcta de roles considerando la mejor cantidad de privilegios
- Hacer uso de las características adicionales del Entra ID como MFA, Acceso Condicional o PIM que puedan en conjunto con IAM/RBAC dar una capa adicional de seguridad y acceso a los recursos sólo cuando se requiera y sea de la identidad que se autentica y autoriza.

3. ¿Qué es la IaC y cuales son sus beneficios?

   Es un marco de trabajo que permite mediante código realizar el aprovisionamiento y configuración de los recursos o servicios. Parte de sus beneficios es que nos permite tener ambientes homologados que no estén sujetos al error humano al momento de hacer el despliegue, también nos permite tener un mayor gobierno de lo que aprovisiona mediante versionamiento, posibilidad de hacer rollback ante cualquier despliegue errado ya que lo podemos integrar con las prácticas de DevOps.

   Dos herramientas son:

   Terraform la cual está orientado al aprovisionamiento de recursos mediante código la cual podemos integrar con control de versiones y tiene soporte multicloud.

   Ansible también lo podemos integrar con herramientas Devops y versiones pero se orienta más a la configuración que al aprovisionamiento.

4. ¿Qué métricas considera esenciales para las soluciones en nube?

   Indicadores de uso que permiten monitorear el uso de Disco, CPU o RAM no sólo para saber si la aplicación es autoeficiente sino también saber si el consumo pagado es el adecuado.

Indicadores de rendimiento para conocer la latencia, el tiempo de atención de solicitudes entrantes y salientes, tasas de error que puedan presentarse en la aplicación.

Indicadores de costo para saber cómo va el consumo actual y conocer si podemos ser más eficientes operativamente y lograr ahorros a la organización.

5. ¿Qué es Docker y cuales son sus componentes?

   Es una tecnología que la podemos usar para la creación y administración de contenedores las cuales las podemos orientar a aplicaciones con arquitectura de microservicios. Sus principales componentes son el servidor Docker (donde se hospeda el Docker demon) y el cliente Docker; ambos conectados por Rest API.

6. Caso Práctico

   Para el presente caso he utilizado la nube de Azure por ser la que más domino y con la que pude participar en más proyectos, además que posee diferentes herramientas que permiten un mejor ordenamiento del proyecto. Por mejores prácticas he dividido cada capa en grupo de recursos para frontend, backend, base de datos y seguridad con monitoreo.

   El usuario que hace uso de la aplicación web lo hará mediante la capa frontend la cual está formado por:

   CND Profile el cual permite la entrega de contenido y la característica de caché para que la aplicación pueda cargar contenido consultado con anterioridad; este componente se respalda de un storage tipo blob en el cual se cargará todo el contenido estático de la web (html, javascript, entre otro contenido)

   Todos los recursos no tienen conexión pública por lo que he utilizado private endpoint para la conectividad entre cada capa y recursos, es por ello que el frontend se conectará al backend mediante los private endpoint.

   Dentro de los backend para una arquitectura orienta a microservicios se utiliza Azure kubernetes services que tendrán zonas de disponibilidad para lograr la contingencia asumiendo que es una aplicación crítica; por temas de seguridad la conectividad usará secretos que serán almacenados en Key Vaults y cada subnet tendrá asociado un Network Security group; para lograr también la automatización de los despliegues mediante DevOps se utilizará un Container registry el cual también sólo se conectará mediante los private endpoints.

   En la capa de base de datos se usará un SQL Database el cual almacenará la cadena de conexión en el Key Vault y los AKS de la capa Backend sólo podrán conectarse haciendo uso de los private endpoint; debido a que el SQL y el AKS están en segmentos diferentes se crea un peering para que permita la conexión y la subnet también tendrá por temas de seguridad su correspondiente Network Security group.

La capa de Seguridad y Monitoreo se podrá usar como soporte para la aplicación el cual contiene el KeyVault y el Monitor, es por ello que el diagrama refleja en cada componente su propio Insights que servirá para revisar la salud de la aplicación.