Reto Técnico – Ingeniero Cloud

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

   Imaginemos que necesitamos alquilar un departamento en un edificio compartido. La **nube pública** es como ese departamento (espacio) el cual es de un proveedor como AWS, Azure o Google Cloud y te ofrece recursos como servidores y almacenamiento, todo esto a través de internet. También te ofrecen que es escalable, económico y perfecto para tus aplicaciones de alta demanda. Pero no tienes el control sobre el hardware y compartes los recursos con otros “inquilinos”.

   Ahora para la **nube privada** imagina que ese edificio es tuyo o alquilado completamente y está dedicado únicamente a tu organización. En esta situación tienes el control para ofrecer seguridad y personalización sobre tus recursos esto es ideal para mantener tus datos sensibles o regulaciones del mercado en el que te encuentras (on-premise). Esto generalmente tiene un coste mayor, tanto en hardware como en mantenimiento y personal humano.

   Además si queremos usar ambas podemos imaginar un puente entre ambos edificios el cual tenemos la llave exclusivamente para acceder a uno y el otro. Mantener ciertos recursos modernos en una nube pública y recursos que deseamos tener el control como la nube privada que puede ser el caso de sistemas legacy pero que podemos brindarles a los clientes acceso mediante sistemas en la nube pública, a esto lo consideramos como **nube híbrida.**

2. Describa tres prácticas de seguridad en la nube.

   En estos tiempos, de continuos cambios y avances tecnológicos debemos enfocarnos en la seguridad de nuestros sistemas y una de las primeras prácticas que debemos implementar es la **encriptación de datos**, tanto en reposo como en tránsito el uso de herramientas para gestionar claves y datos necesarios. El uso de herramientas como Hashicorp Vault para on-premise o Azure Key Vault en la nube privada nos ayuda a proteger esta información.

   El **control de identidad y acceso** es una parte fundamental ya que nos ayuda a poder identificar y dar acceso al personal sobre los recursos. El uso de Cyberark o Azure Active Directory que ayudan a gestionar ese control y permite dar el mínimo privilegio para para recursos de ambientes que no necesitan que otro equipos se incluyan, por ejemplo desarrolladores no deben tener acceso a producción.

   Y una tercera buena práctica es el **monitoreo y auditorías** de todos los recursos, herramientas como Qualys, Fortify que identifican vulnerabilidades de software o Azure Security Center que nos ayuda a responder rápidamente ante amenazas detectadas.

3. ¿Qué es la IaC, y cuáles son sus principales beneficios?, mencione 2 herramientas de iaC y sus principales características.

La IaC nos permite definir servidores, redes y servicios como código es decir nos permite mantener un plano detallado y de cómo se define nuestra infraestructura en algún proveedor Cloud.

Uno de los beneficios es **evitar errores humanos** ya que al hacerlo manual podrían configurarse de distintas formas los servidores.
Podemos desplegar las veces que queramos los recursos únicamente usando unos pocos comandos de esta manera teniendo **escalabilidad**.
Al usar Git controlamos las **versiones**.
Y para agilizar este proceso la integración de CI/CD nos ayuda a tener **despliegues continuos**.

Para el caso de infraestructura en la nube es muy productivo el uso de **terraform** el cual podemos crear vpc, subredes, instancias, etc. De la misma manera mantiene la gestión de estados y funciona con varios servicios en la nube.

Ansible en el aprovisionamiento de IaC es funcional en la ejecución de tareas, configurar servidores y desplegar aplicaciones, su simplicidad de lectura y su gestión en las tareas lo vuelven de gran importancia.

El uso de ambas son se vuelve herramientas poderosas que permiten recrear la infraestructura en caso de desastres.

4. ¿Qué métricas considera esenciales para el monitoreo de soluciones en la nube?

Considero que una de las principales métricas es el **rendimiento** donde podremos analizar y actuar sobre los recursos que están consumiendo más de lo necesario en CPU, memoria, disco o red, las herramientas que podemos usar son Prometheus + Grafana y Azure Monitor.

La **disponibilidad** de los servidores y servicios que estén respondiendo oportunamente ya que es esencial para cumplir con los estándares y SLAs establecidos, esto se logra con herramientas como Nagios y Application Insights para detectar errores en aplicaciones.

Y como punto importante la **seguridad** el cual nos permitirá encontrar los intentos de accesos no autorizados o cambios en configuraciones de red críticos, podemos usar herramientas como Elasticsearch + Kibana y Security Center en todo el ecosistema Azure.

5. ¿Qué es Docker y cuáles son sus componentes principales?

Docker es una plataforma que permite gestionar contenedores, estos son pequeñas máquinas virtuales que contienen todo lo necesario(librerías, variables de entornos, SO base) el cual empaca la aplicación para que funcione en cualquier lugar.
De esta manera podemos tener una pequeña VM pero sin el peso que normalmente conlleva y mantiene la aplicación en un entorno aislado.

Los componentes principales son las **Docker Images** el cual nos permite crear una instantánea de una aplicación lista para su ejecución.
Docker Container es donde podemos ejecutar una imagen de forma efímera.
Docker registry, aquí podremos almacenar en la nube las imágenes y poder usarlas las veces que sean necesarias. De la misma forma podemos tenerla de manera privada para la organización y publicarlas en Azure Container Registry o Amazon Elastic Container Registry.

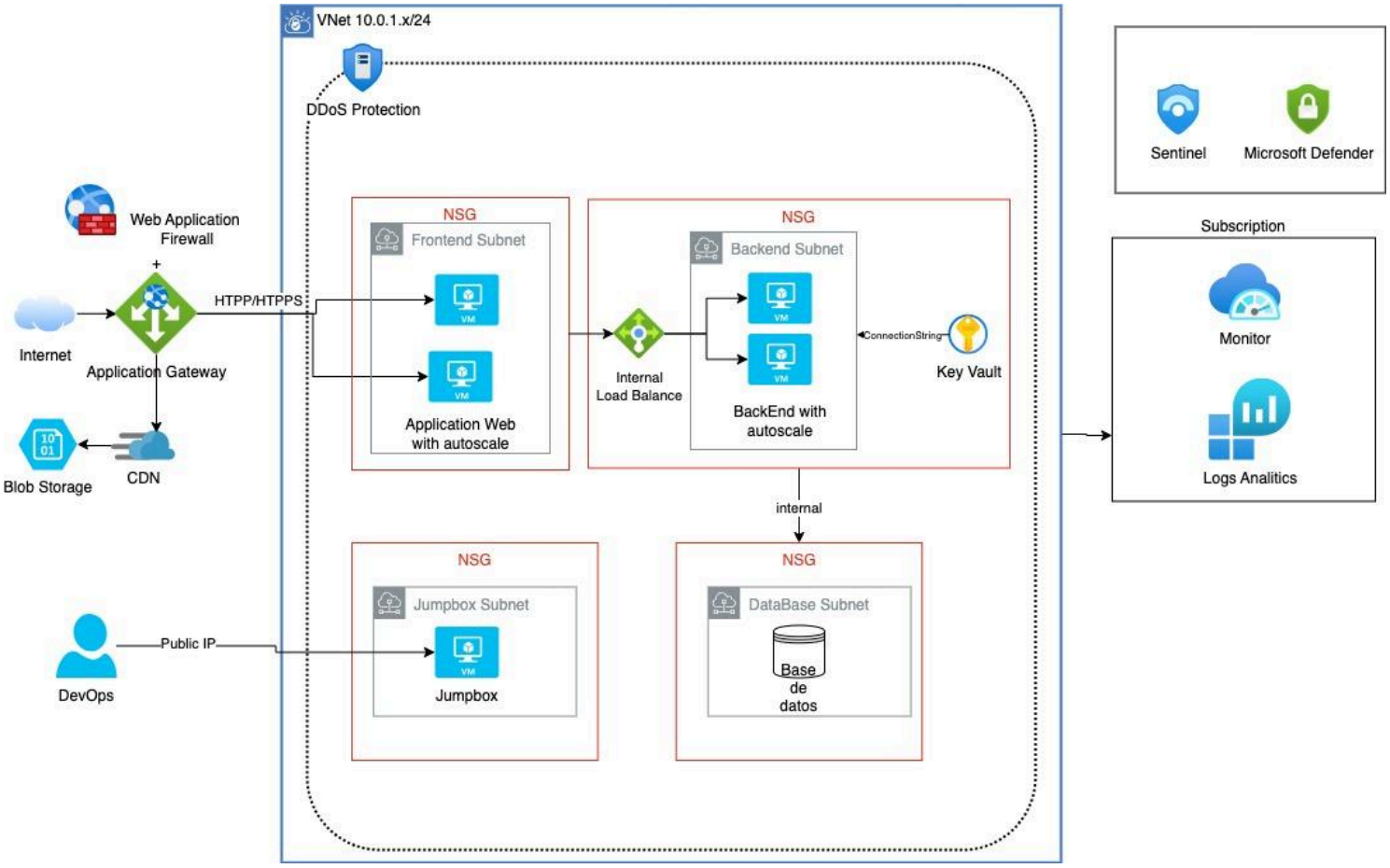
De esta manera podemos tener consistencia en el despliegue de nuestras aplicaciones, portabilidad para poder usarlo en el SO que queramos y eficiencia e integración.

6. Caso Práctico:

Cree un diseño de arquitectura para una aplicación nativa de nube considerando los siguientes componentes:
Frontend: Una aplicación web que los clientes utilizarán para navegación.
Backend: Servicios que se comunican con la base de datos y el frontend.
Base de datos: Un sistema de gestión de base de datos que almacene información.
Almacenamiento de objetos: Para gestionar imágenes y contenido estático.

Continúa en la siguiente página.

Diagrama de arquitectura propuesto



Continúa en la siguiente página.

Para la arquitectura propuesta se basa en **Azure** como proveedor de servicios en la nube debido a que Microsoft ha sido un gestor de seguridad que ha evolucionado tecnológicamente y ofrece una combinación sólida de redes seguras y herramientas de seguridad integradas como son Microsoft Defender y Sentinel.

Dentro del diseño he optado por agregar recursos de seguridad esenciales para un ecosistema robusto y seguro como el caso de **WAF** nos ayuda a mitigar amenazas OWASP, SQLi, etc.

Usando **CDN y Blob Storage** resuelve la entrega eficiente de contenido estático y almacén de caché, reduciendo la carga en los servidores frontend y backend. Estos componentes se mantienen aislados de la VNet debido a que ya cuentan con mecanismos que controlan el acceso a datos y redes virtuales pero podría incluir más mecanismos de seguridad.

Un **Application Gateway** que nos permitirá equilibrar la carga de tráfico web además de aplicar reglas de enrutamiento para un mejor control de rutas además de **DDoS Protection** que nos ayudará a absorber los ataques DDoS.

Se definen **subredes** dedicadas para frontend, backend y base de datos, cada una protegida por un **Network Security Groups**. Esto permitirá aplicar reglas de seguridad que bloqueen el tráfico no autorizado y se limiten las comunicaciones a los recursos únicamente necesarios.

Application Gateway enruta las solicitudes hacia el frontend el cual envía las solicitudes hacia el backend directamente teniendo en cuenta que las subnets evitan que se exponga hacia internet.

Load Balance interno el cual no expondrá IPs públicas, de esta manera ayuda a una alta disponibilidad, escalabilidad mejorando la seguridad y rendimiento.

La disponibilidad de un recurso de seguridad como lo es **Key Vault**, accesible sólo mediante identidades administradas, de esta manera ayudará a mantener cadenas de conexión y claves seguras.

La subnet dedicada para **backend** evita accesos no autorizados y las reglas de la **NSG** ayudará a bloquear tráfico de internet.

La **Base de Datos** se encuentra con su propio **subnet dedicada** y junto a las reglas de NSG ayudará a que únicamente los recursos del backend y administradores puedan acceder a ellos.

Para mantener métricas de monitoreo el uso de **Azure Monitor** y **Log Analytics** es necesario y así podemos recopilar métricas de VMs, base de datos y aplicaciones además de mantener alertas proactivas para detectar anomalías.

Y el uso indiscutible de los mecanismos de análisis de seguridad y garantizar el compliance de una organización esto lo podemos lograr con la suite de **Microsoft Defender** y **Sentinel** el cual nos proporciona visibilidad total del entorno, usando las herramientas incorporadas podremos identificar patrones sospechosos, mitigar vulnerabilidades en los recursos responsables, mitigar incidentes, análisis de logs y cumplimiento de estándares.

Tener aislamiento de accesos es primordial para minimizar el acceso no autorizado, es por ello que se usa un **jumpbox** que servirá de enlace y conexión hacia los servidores y servicios de azure así evitaremos accesos directos desde nuestro dispositivo.

Este enfoque depende mucho de las necesidades del negocio, como recomendación podría agregar **Azure Front Door** para el enrutamiento global para usuarios finales, sólo si el caso de este requerimiento conlleva un flujo alto fuera de una zona.