.NET Microservices – Azure DevOps and AKS

Section 15: Azure API Management - Notes

1. Introduction to Azure API Management (APIM)

Azure APIM enables secure, reliable, and scalable access to services by providing features such as:

- **Traffic Management**: Routes and manages API traffic, balancing requests among services and scaling automatically.
- **Security**: Offers built-in policies to enforce security, including rate limiting, IP filtering, and CORS. APIM also supports authentication via OAuth2, Azure AD, and JWT token validation.
- **Analytics**: Provides insights into API usage, errors, and response times, essential for monitoring and improving API performance.
- **Developer Portal**: A customizable portal for developers to discover, subscribe to, and access APIs with documentation and testing tools.

2. How APIM Works Internally with AKS as Backend

In a setup where APIM is acting as a frontend for services running on Azure Kubernetes Service (AKS), the internal workflow looks like this:

1. **Request Routing**: Client applications (such as mobile apps or web clients) send requests to the APIM endpoint.
2. **Request Processing**:
    - APIM verifies the request using applied policies, such as validating tokens or checking IP restrictions.
    - It may rewrite, transform, or rate-limit the request based on configured rules.
3. **Backend Communication**:
    - After verification, APIM routes the request to the appropriate AKS service. APIM uses a **backend configuration** that defines which AKS service should handle the request based on rules.
    - For this, the AKS service's IP address and port are typically registered in APIM as a backend service, often with DNS or a Virtual Network (VNet) for secure communication.
4. **Response Processing**:
    - APIM receives the response from the AKS service.
    - Any post-processing policies (e.g., transforming the response format or stripping sensitive information) are applied before sending the response back to the client.

This setup provides an added layer of abstraction and control over how client applications interact with backend services on AKS, creating a structured API ecosystem with policies that enforce security, logging, and monitoring.

3. Benefits of Using APIM as a Gateway for AKS Services

- **Security Control**: Acts as a security gateway for AKS services, enforcing authentication and access control.
- **Unified Interface**: Provides a single entry point for multiple services running in the AKS cluster, simplifying client access.
- **Resilience and Scalability**: APIM scales requests across backend AKS services, handling load management and retry policies.
- **Rate Limiting and Throttling**: Manages traffic spikes and prevents backend overload by controlling request limits.
- **Analytics and Logging**: Tracks API usage and performance metrics, helping diagnose issues and optimize services.

Key Terminologies in APIM

- **Product**: Grouping of APIs that can be exposed to developers with specific policies.
- **Policy**: A set of rules applied to API requests and responses, like caching, authorization, or logging.
- **Backend**: A configuration in APIM representing the backend service, like an AKS microservice, with settings for routing and authentication.
- **Operations**: Defines the various methods (GET, POST, etc.) for each endpoint in an API.