What API Gateway should I use?

Ocelot

I started off by trying out Ocelot. Since by backend is in c#, it was very accessible to try-out Ocelot and it works fine locally. But as soon as you want to containerize it on docker you must take many additional steps to configure environmental variables which is very inconvenient. Instead, it might be more valuable switching before I'm too deep into it.

Ocelot is also one of the less popular gateways which leads to less community support and sometimes even outdated documentation. It is more popular in general to use API gateway tools since many features are already built in. Therefore, I will start looking for API Gateway tools rather than frameworks.

API Gateway Comparison

A comparison on API Gateways according to (Witts, 2024)

Feature/Platform	Amazon API Gateway	Azure API Management	Boomi API Management	Google API Gateway	IBM API Connect	Kong Gateway	MuleSoft Anypoint Flex Gateway	WSO2 API Manager
Deployment Environments	Cloud-native, serverless, containerized	Multicloud, hybrid, on-premises, cloud	Cloud, on-premises, edge	Google Cloud	Cloud, on-premises, hybrid	Cloud, on-premises, Kubernetes	Cloud-native, containerized, CI/CD integrated	Cloud, on-premises, hybrid, open-source
Supported API Types	RESTful, WebSocket	RESTful, GraphQL, SOAP, WebSocket	RESTful, SOAP	RESTful	RESTful, SOAP	RESTful, GraphQL	RESTful, GraphQL	REST, GraphQL, AsyncAPIs, WebSocket
Security Features	IAM, CORS, web firewall, API keys	OAuth2, JWT, IP filtering	Third-party authentication	API keys, GCP service accounts, Google ID tokens	OAuth 2.0, OpenID Connect, JWT, threat management	Out-of-the-box plugins, custom plugins	Zero-trust, identity and access policies	OAuth2, fine-grained policies, threat protection
Pricing Model	Pay-per-use, tiered pricing	Scalable pricing, various service tiers	Subscription-based	Pay-per-use	Subscription-based	Open-source and enterprise plans	Subscription-based	Open-source and enterprise plans
Developer Portal	No native portal	Customizable developer portal	Included developer portal	Integrated with Google Cloud Platform	Branded developer portal	Community-supported developer resources	Integrated developer portal	Open-source, API marketplace
Performance & Scalability	Low-latency via Amazon CloudFront	High scalability with global reach	Real-time integrations	High scalability with Envoy	Centralized operations for large-scale environments	Lightweight, >50k TPS per node	Supports large-scale microservices	Scalable, with integrations for government and enterprise
Monitoring & Analytics	Latency, error rates, API version management	Logs, metrics, quotas, comprehensive observability	Historical and real-time API health insights	Usage metrics, logging, tracing	Real-time transaction visibility	Plugins for analytics, extensive community support	Full traffic management policies	Advanced logging, monitoring, analytics
Unique Strengths	Integrates with AWS services, cost-effective	Hybrid/multicloud support, security compliance	Simple deployment, integration with Boomi ecosystem	Built-in Google Cloud support, high security	Centralized management, powerful AI/automation capabilities	Open-source, highly adaptable, DevOps & GitOps integration	CI/CD pipeline integration, high adaptability	Open-source, highly customizable, extensive protocol support
Ideal For	High-traffic apps with AWS integration	Enterprises needing robust security and hybrid/multicloud solutions	Businesses needing ease of use and integration flexibility	Organizations on Google Cloud needing simple API management	Organizations requiring comprehensive, secure API management	Developers seeking flexible, open-source API management	Teams using DevOps/CI/CD workflows, high-performance APIs	Enterprises with extensive customization needs

What am I looking for in an API Gateway?

Features

Musts

- Rate Limiting
- Server Discovery
- Authorization
- Authentication

Nice to have

- Load Balancing (I can always use a separate load balancer behind the gateway)

But it appears that all the gateways provided the musts.

Non-Functional Requirements

- Scale up to 1 million concurrent users
- Low Latency is preferred
- Protect website against DDoS attacks (Security: LO6)
- Setup firewall for website (Security: LO6)

Using the documentation of each API gateway as sources I managed to narrow down the decision on my API gateway. Also important to mention is that I don't have any budget: (Amazon, n.d.), (Amazon, n.d.), (Microsoft, 2024), (Microsoft, 2024)

Non-Functional Requirements	Amazon API Gateway	Azure API Management	Google API Gateway	IBM API Connect	Kong Gateway	MuleSoft Anypoint Flex Gateway	Boomi API Management	WSO2 API Manager
Scale up to 1 million concurrent users	Yes, designed for high concurrency	Yes, scales with additional units (not in Developer tier)						
Low Latency	Yes, optimized for low latency	Yes, optimized for performance						
Protect data in a GDPR-compliant manner	Yes, offers encryption, IAM policies, and integration with AWS services for compliance.	Yes, data encryption and IAM						
Protect website against DDoS attacks	Can integrate with AWS shield	Yes, DDoS protection						
Setup firewall for website	Can integrate with AWS WAF	Yes, firewall options available						
Auto Scaling	Yes, AWS scales automatically	Yes, add/remove scale units						
Pricing Model	Pay-per-use, tiered pricing Educational Account	Scalable pricing, various service tiers Fontys Credits	Subscription-based	Pay-per-use	Subscription-based	Open-source and enterprise plans	Subscription-based	Open-source and enterprise plans

So, basically AWS and Azure are both very similar in their API gateway. Before choosing what cloud platform to choose, it might be useful to consider if I need the platform for anything else so I can research that too, but in the case of the API Gateway it doesn't matter which one I choose.

Bibliography

Amazon. (n.d.). *Amazon API Gateway FAQs | API Management | Amazon Web Services*. Retrieved from Amazon Web Services, Inc.: https://aws.amazon.com/api-gateway/faqs/#:~:text=Amazon%20API%20Gateway%20acts%20as,of%20traffic%20your%20API%20receives.

Amazon. (n.d.). *What is Amazon API Gateway? - Amazon API Gateway*. Retrieved from https://docs.aws.amazon.com/apigateway/latest/developerguide/welcome.html#api-gateway-overview-features

Microsoft. (2024, October 3). *API gateway overview*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/azure/api-management/api-management-gateways-overview

Microsoft. (2024, April 3). *Feature-based comparison of the Azure API Management tiers*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/azure/api-management/api-management-features

Witts, J. (2024, July 8). *The top 8 API gateways*. Retrieved from Expert Insights: https://expertinsights.com/insights/the-top-api-gateways/