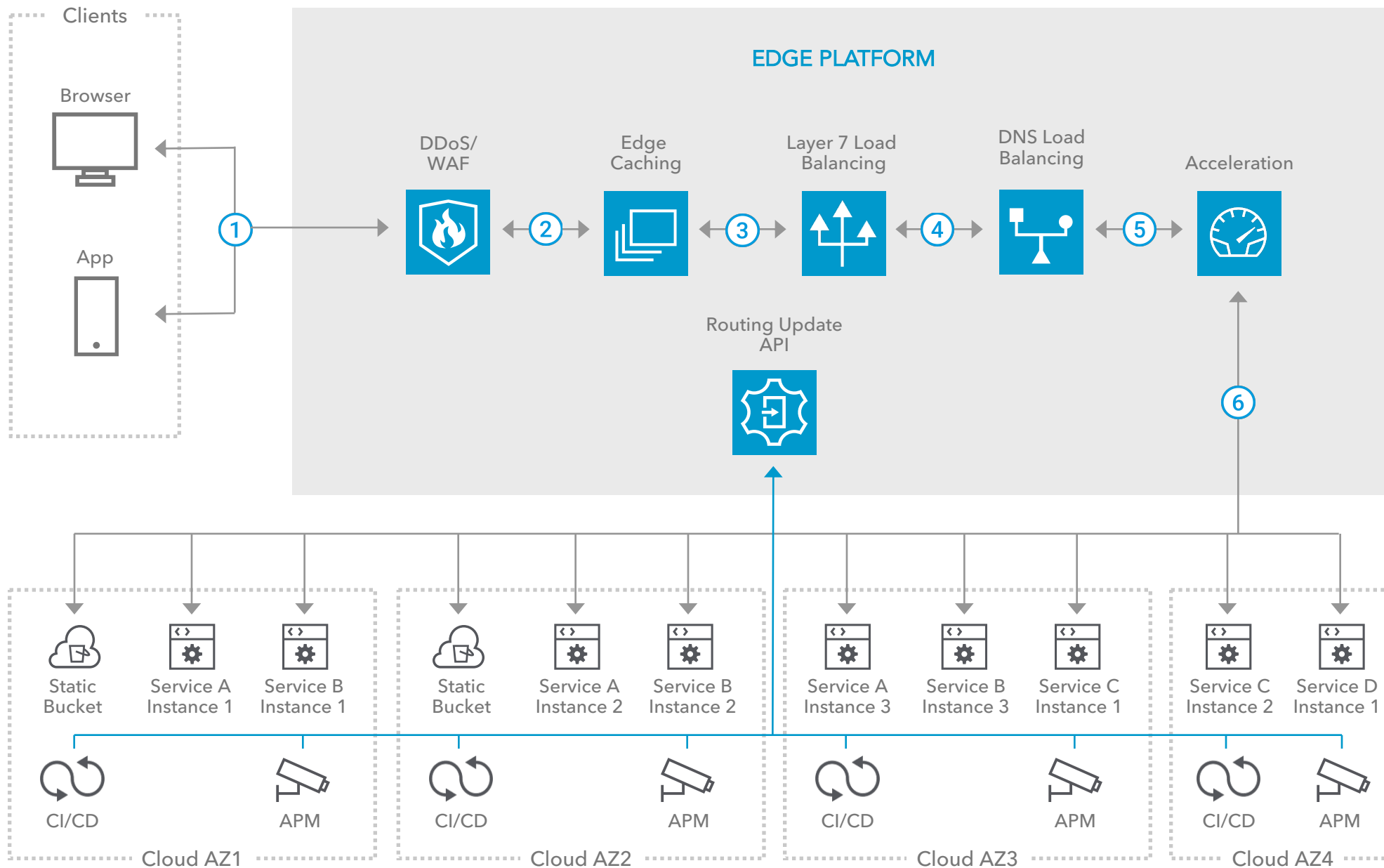


MICROSERVICES TRAFFIC MANAGEMENT

Reference Architecture



OVERVIEW

Modern applications rely on microservices to improve application performance and reduce development time. These benefits come at the cost of increased complexity in managing and securing the numerous services that make up the app. Furthermore, implementing a consistent set of security controls becomes unmanageable if each endpoint is dealt with individually. The edge is the solution to that complexity. Akamai can consolidate security, delivery, and load balancing functions in a single platform. Doing so allows developers to build and iterate applications instead of reworking security and operational requirements. IT decision-makers also benefit from the decreased vendor lock-in, as centralizing these functions gives them more flexibility around which cloud provider to choose.

- 1 Security controls (WAF, geoblocking, rate limits, reputation, and bot detection) applied to request at edge.
- 2 Static page assets (images, videos, JS, CSS) served from local user's edge cache.
- 3 Layer 7 routing allows traffic to be directed to individual microservices. For example:
/chat -> Ashburn + Oregon
/login -> Ashburn + Dublin + Frankfurt
- 4 Global load balancing distributes traffic across active instances of compute (for microservices) or static buckets (for cacheable content).
- 5 Route, TCP, and TLS optimizations between client and cloud edge maximize availability and minimize overall response time for non-cacheable requests.
- 6 Traffic delivered over public internet from known Akamai IP ranges or established private connectivity.

KEY PRODUCTS

DDoS/WAF ► App & API Protector

Edge caching and acceleration ► Ion/API Acceleration

Layer 7 load balancing ► Application Load Balancer or PM behaviors

DNS load balancing ► Global Traffic Manager