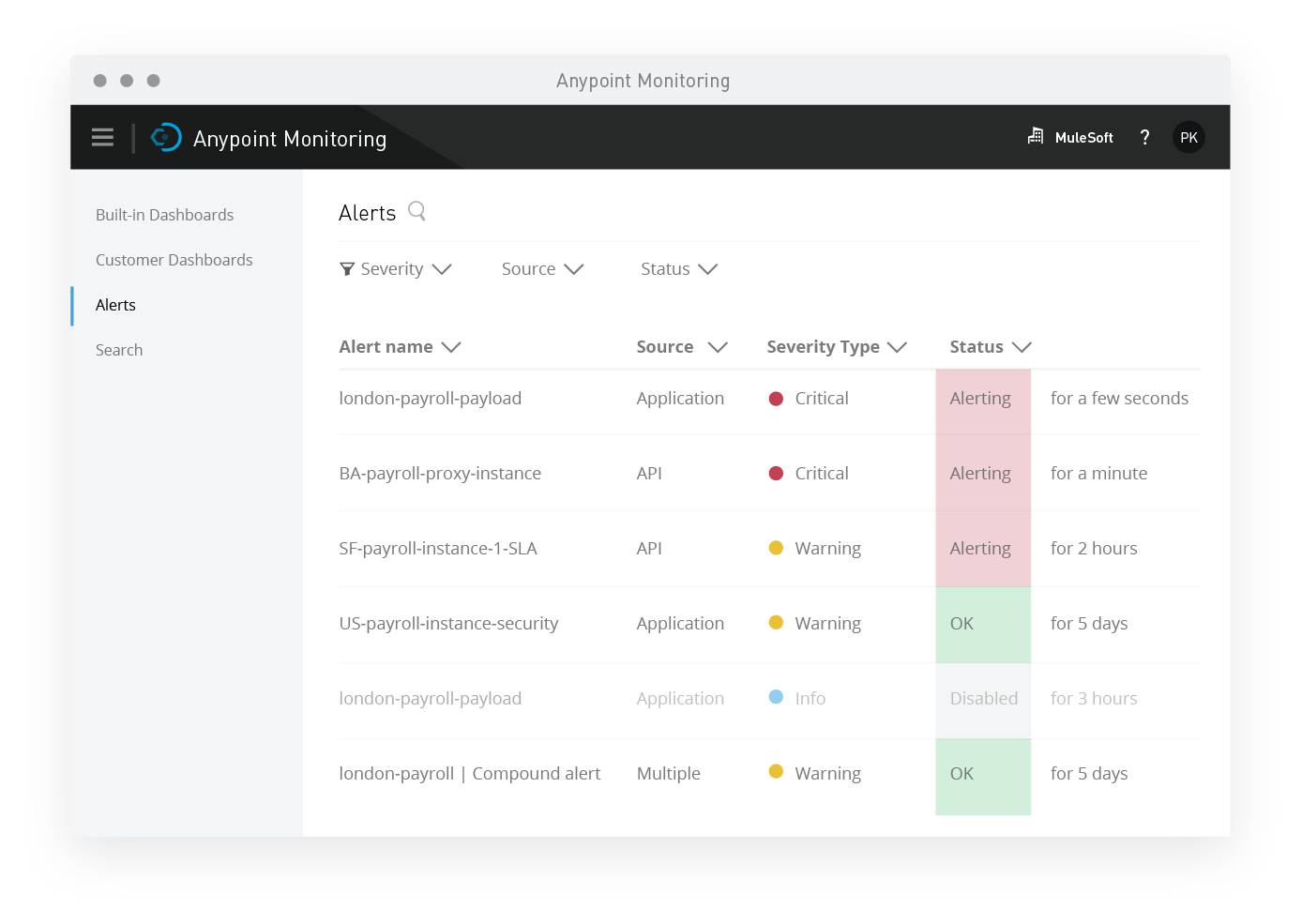
# API Monitoring

API monitoring solutions are needed to monitor API health and uptime and to trigger alerts when an API is down or in the case of returned errors. An API Gateway platform should come with basic API monitoring built in, and most of the gateways do have such capabilities. Basic API health check monitoring is built into all major API platforms, such as AWS, Azure, Mulesoft, Kong, 3Scales, and others.

An example API monitoring dashboard from Mulesoft that quickly shows API uptime is shown below:



However, besides basic health check monitoring, it is important for any organization to also monitor API latency and performance, as well as to have ability to trigger monitoring alerts based on specific HTTP codes, performance thresholds, or security-based triggers.

API Monitoring Capabilities

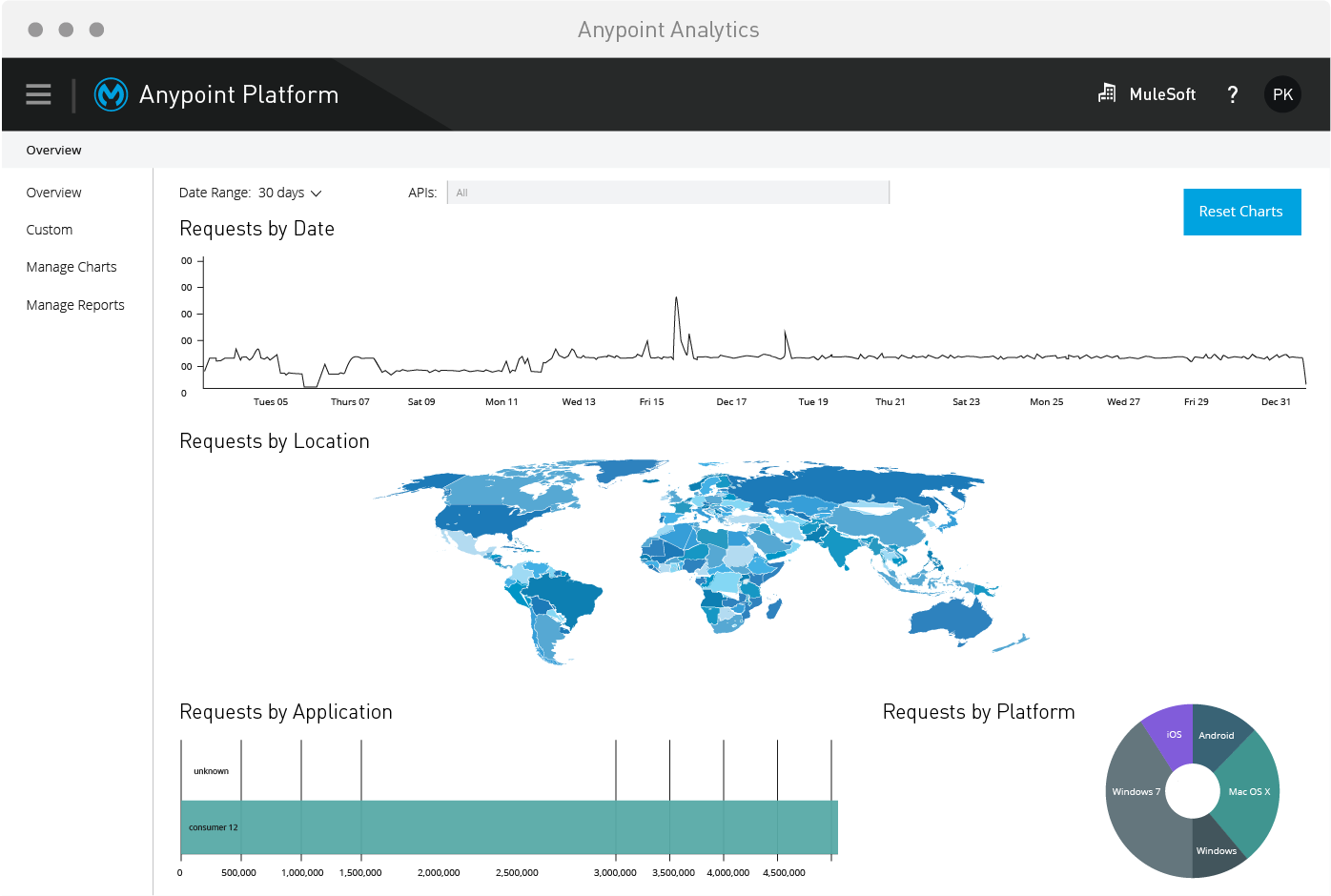
API monitoring capabilities include a wide range of monitoring and alerting capabilities. An API gateway typically doesn’t provide all of such capabilities and requires additional monitoring tools and platforms. When selecting an API gateway, it is important to review provided monitoring capabilities and native integrations with external monitoring tools.

As mentioned in the previous section, all API gateways provide basic health check to quickly determine whether an API is up and responding to incoming requests.

Besides health check monitoring, other monitoring capabilities to consider include:

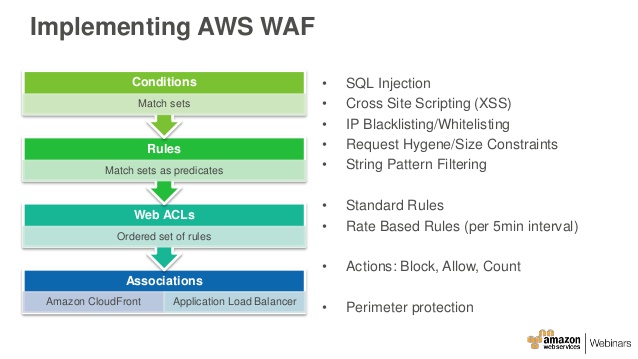
* **Usage Monitoring**

Usage monitoring cover real-time and historical usage statistics by date, application, platform, and geographic region. An example such dashboard is provided by Mulesoft Anypoint Platform:

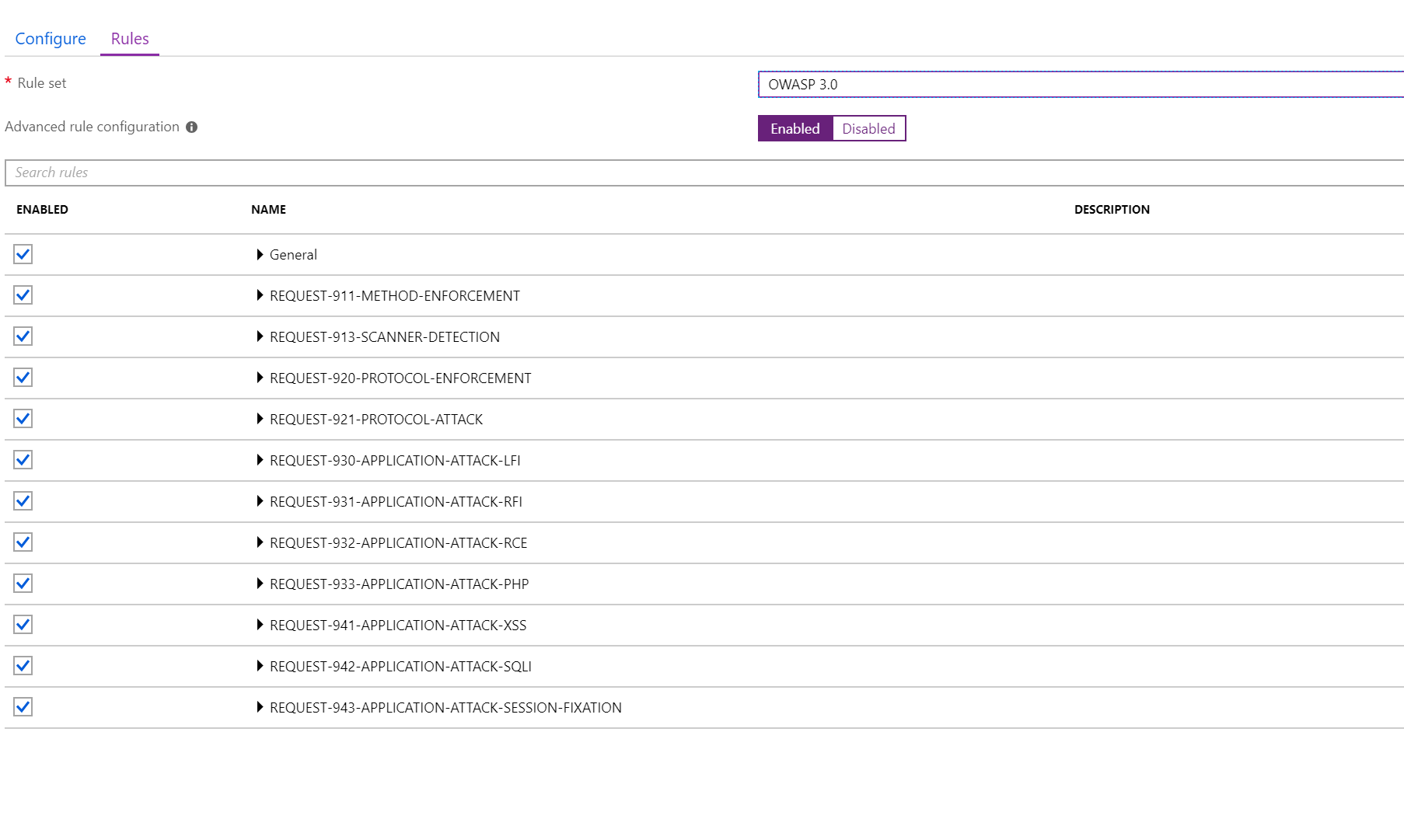


* **Security Monitoring**

Security monitoring provides security alerts on malicious attempt to inject data into an API request payload (e.g., SQL or Javascript injection), execute a DDOS attack against an API, or attempt to hack into the credentials required by a given API. Basic API gateway capabilities are limited to alert on unauthorized API request attempts that throw an HTTP 401 or 403 codes. However, DDOS prevention, injection prevention, and other malicious attempts are not detected by any of the provided API gateways without a separate security appliance such as a Web Application Firewall (WAF), which can execute a series of OWAS 2.2.9 and 3.0 rules and trigger security alerts. Therefore, as a best practice, all API Gateway traffic is trafficked through a WAF firewall as the first line of defense. Security monitoring, therefore, is extended to both WAF Application Gateway and an API Gateway. Below is a sample workflow process of an AWS WAF which should be positioned in front of an AWS API Gateway:



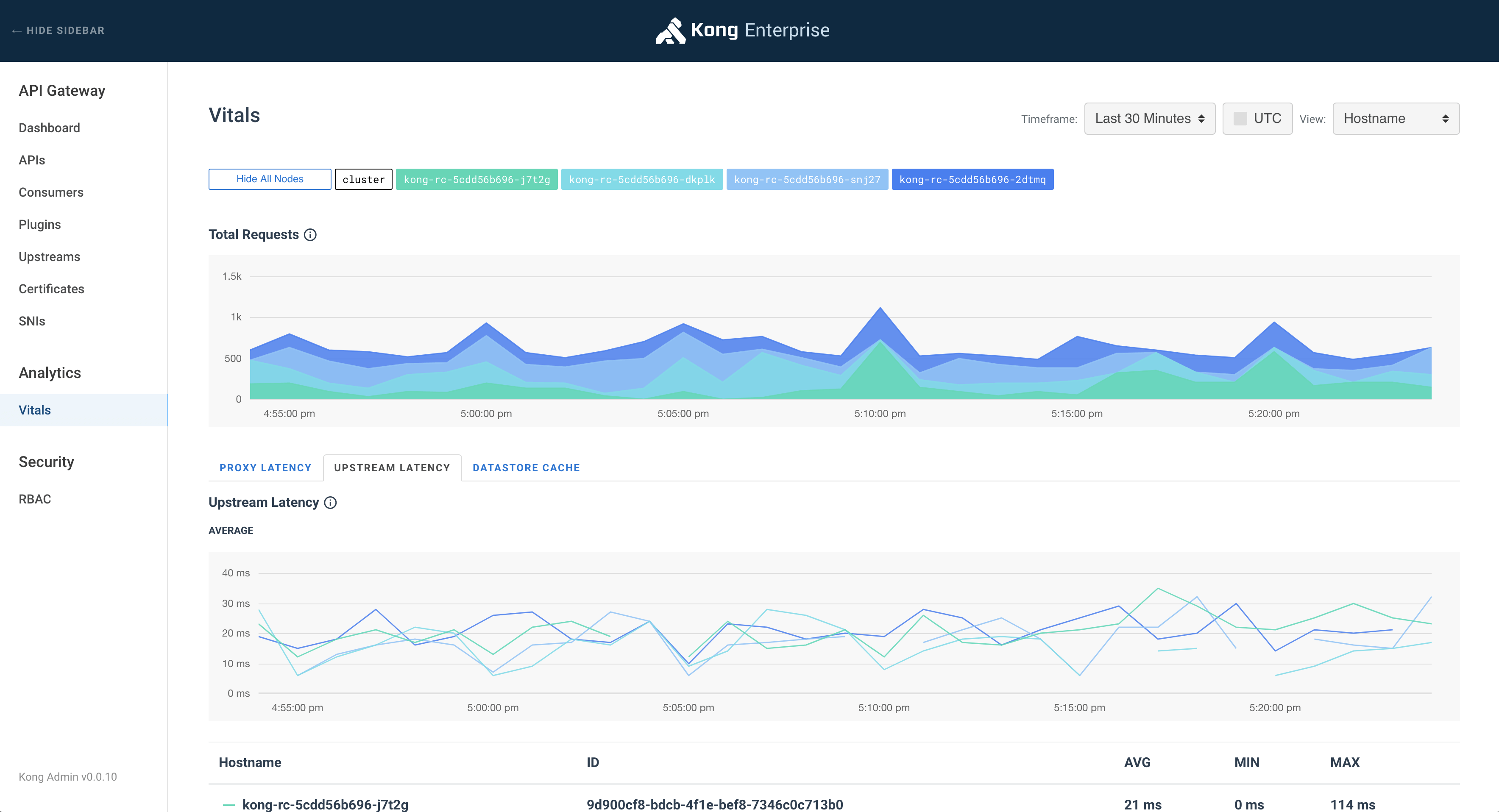
Here is a sample Azure Application Gateway / WAF configuration to protect APIs from malicious injection and DDOS attacks:



* **Performance Monitoring**

Performance monitoring of APIs is usually a built-in feature of an API gateway. Enhanced performance monitoring is provided by third-party performance monitoring tools such as New Relic and AppDynamics. An API performance monitor would show total API requests by operation, latency per each operation, highlighted operations and calls that took longer than expected.

An example API performance monitoring dashboard is shown below for Kong Enterprise API platform:



Monitoring Alerting and Notifications

As a best practice, a number of API monitoring alerts should be configured on each critical API, that is essential to a given application. Alerts are tied to monitoring types:

* Uptime Alerts are triggered when an API is not responding to requests.
* Usage Alerts are triggered when a certain API is accessed more than a predefined limit. For example, a usage alert can be set up to fire when an API is requested more than 100 times in a minute.
* Security Alerts are triggered when an API returns 401 or 403 security response codes, when a malicious attack is detected based on security rules, or when a network security attack such as DDOS is detected by the network monitor for the API traffic.
* Performance Alerts are triggered when an API doesn’t perform per SLAs. SLAs can be configured for the alerting, for example, to trigger an alert if an operation within an API takes longer than expected to return the results.

An API management platform should provide capabilities to send alerts via email, SMS, or pager to a corporate support department. Certain API platforms rely on third parties such as ServiceNow or PagerDuty to perform such notifications. Mulesoft and Kong provide integrations with ServiceNow to be able to create a support ticket in ServiceNow based on a detected issue. AWS relies on its other solutions such as CloudWatch to do alert management and notifications, and AWS API Gateway provides built-in integration with CloudWatch.

As a best practice, it is important to add alerts to all core APIs that are essential to the quality of a given application. Uptime, security, performance, and usage alerts should be configured on each core API. SLAs for performance should be defined based on a specific API operation, since some API operations should take longer than others, depending on their backend integration and processing logic.

Below is a sample configuration for setting a new AWS CloudWatch Alarm to be fired when a specific event, such as reaching API max request threshold, occurs:

