Secure, control and observe every call to your APIs

Rate limits, identity with OpenID Connect, LDAP, SAML, Keycloak, JWT, API keys, HMAC and mutual TLS, access rules and traffic management for every API, managed in one place. Analytics and request records stay in your own data stores. Every rule is a plugin with its own form in the console; attach it to a route, a service or a consumer, or apply it gateway-wide.

Capabilities

Rate limiting
Limit requests, request rates and concurrent connections by client, Consumer or a key you choose, with counters shared across gateway nodes in a Redis you provide.

Identity
OpenID Connect, LDAP, SAML, Keycloak, JWT, API keys, HMAC and mutual TLS, and basic authentication; several methods on one Route with multi-auth.

Access rules
Allow or deny by IP address or User-Agent, CORS rules, URI blocking, and Routes restricted to the Consumers or groups you name.

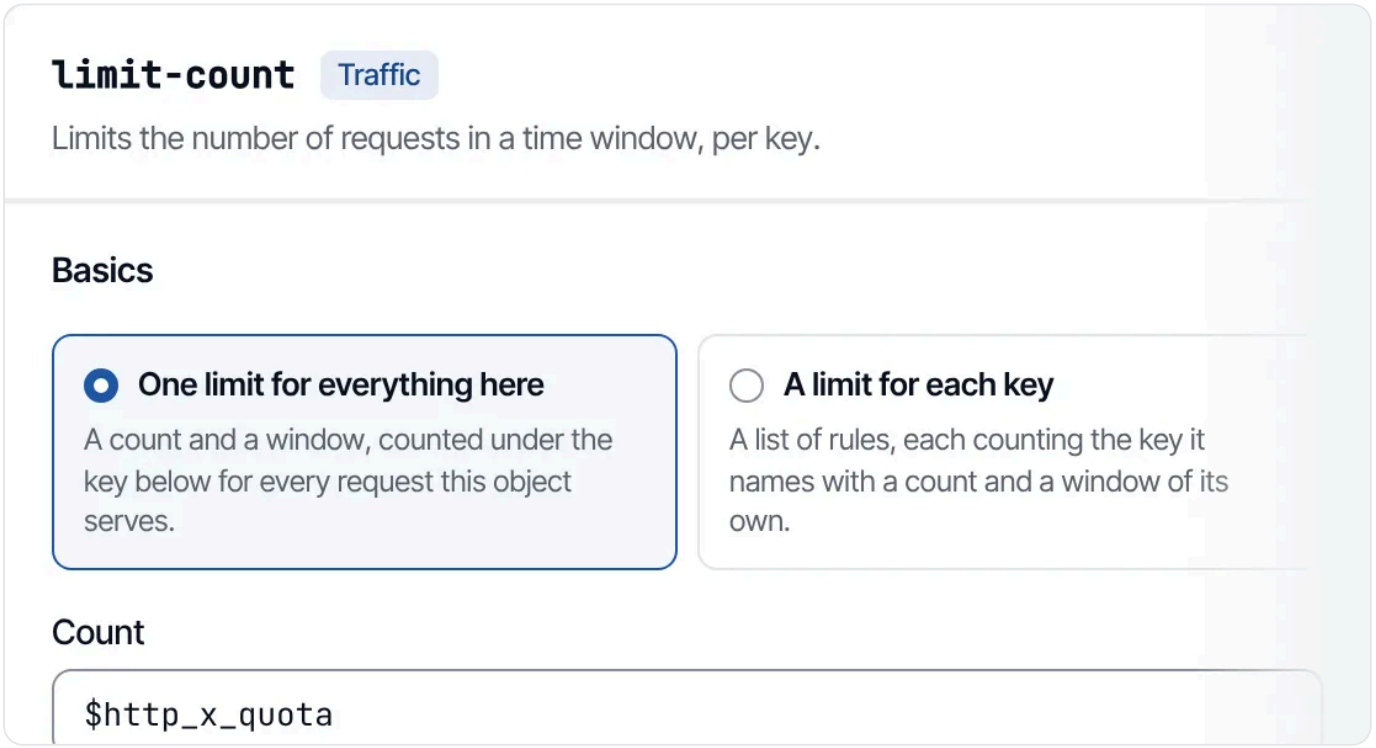
Transformation
Rewrite requests and responses on their way through the gateway, including gRPC transcoding.

Caching
Cache upstream responses on disk or in memory, keyed on the parts of the request you choose.

Traffic management
Split traffic between upstream versions for canary releases; stop sending requests to a failing backend with a circuit breaker; health checks take nodes out of rotation.

Request validation
Refuse requests that do not match a JSON schema or your OpenAPI document before they reach the backend.

Protocols
HTTP/1.1, HTTP/2, gRPC and WebSocket on Routes; TCP, UDP and MQTT on stream Routes.



The limit-count form on a Route, drawn from the engine's own schema: one limit for everything or a limit per key, and a count read from the request.

Observe: analytics, request records and alerts from your own stores

Analytics

Requests, status share, p50 and p95 response times, top Routes, Upstreams and Consumers, gateway nodes, and the changes made in the window drawn on the same chart, read from your Prometheus.

Traffic Explorer

One record per request with its status, Route, Consumer, timings and the headers you chose to record, read from your Elasticsearch or OpenSearch. Secrets are masked before a record is stored.

Alerts

Rules on error rate and response time per Route, Service or Upstream, sent by email or signed webhook; history kept 90 days.

One console for every gateway cluster

Environments

Each Environment is one gateway cluster with its own engine, data stores and members. Copy a Route from Staging to Production with a preview of what changes.

Roles from your directory

Console sign-in with local accounts or Microsoft Active Directory; directory groups map to roles, and a role can be limited to one Environment.

An audit log of every change

Every change is written as an audit record before it is applied; a refused change is a record too. Each record carries the actor, the object before and after and the result; kept 365 days by default, exported as CSV or a log line for your SIEM.

Automation

The management API is documented with OpenAPI; service account tokens let CI/CD pipelines change gateway configuration under the same roles and the same audit log.

Plugin catalog

110 Plugins, each with a form drawn from the engine's schema and a Raw JSON tab. The ones an API team looks for first:

`limit-count` `limit-req` `limit-conn` `key-auth` `jwt-auth` `openid-connect` `ldap-auth` `saml-auth` `authz-keycloak` `hmac-auth` `basic-auth` `multi-auth` `opa` `ip-restriction` `cors` `csrf` `uri-blocker` `consumer-restriction` `request-validation` `proxy-cache` `proxy-rewrite` `response-rewrite` `body-transformer` `data-mask` `traffic-split` `api-breaker` `grpc-transcode` `mocking` `fault-injection` `elasticsearch-logger` `kafka-logger` `splunk-hec-logging` `syslog` `prometheus` `zipkin`

NEXT STEP

See the forms, Analytics and Traffic Explorer on traffic like yours. Request a demo at g8.dev/request-a-demo or write to hello@g8.dev. Read next: G8 API Security Datasheet (2 pages), at g8.dev/downloads.

`g8.dev/request-a-demo · hello@g8.dev`