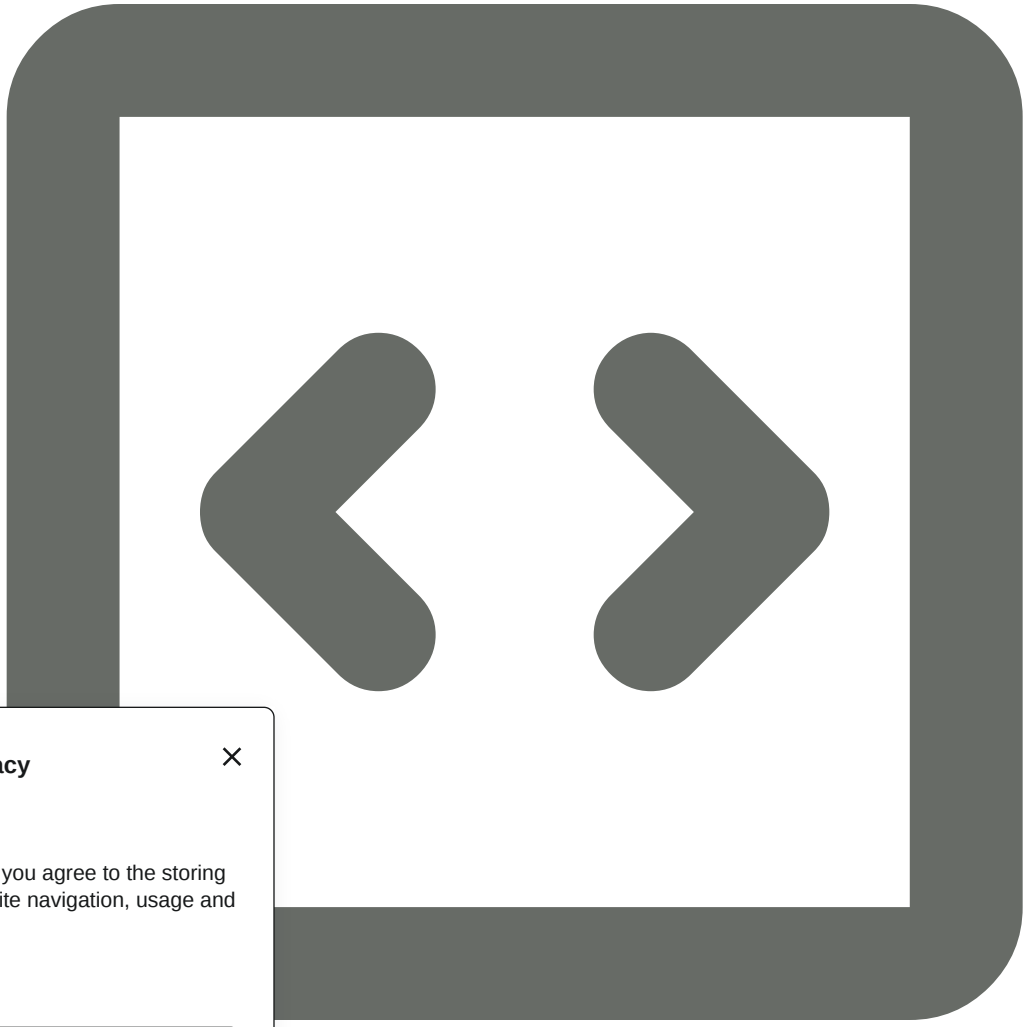
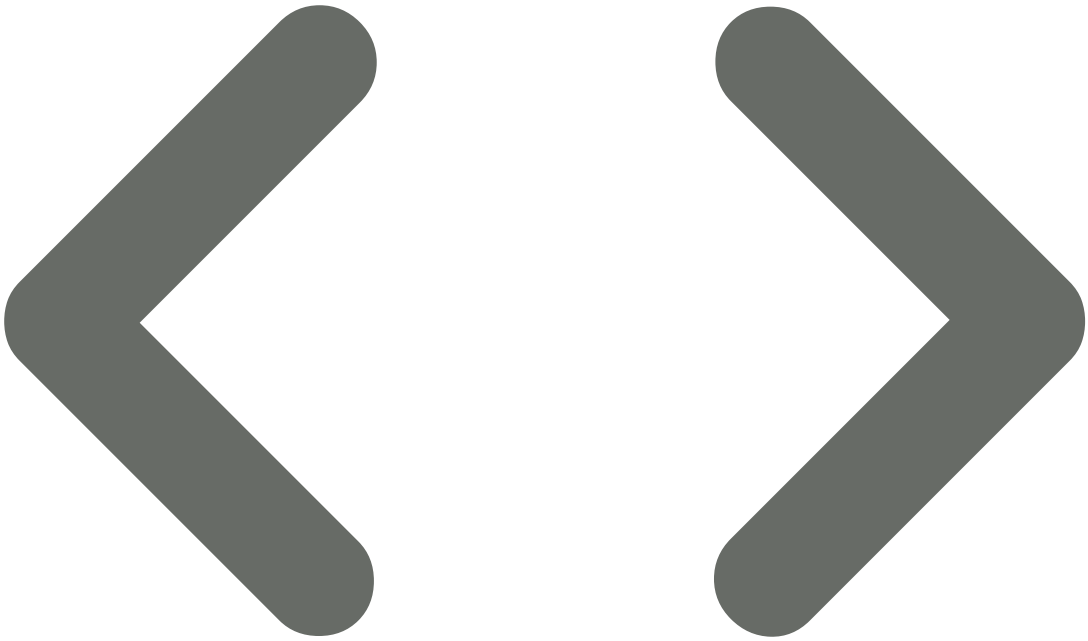
SearchCommand or control key
⌘ / ctrl
K key
K
Ask AI
Kong
- Platform
Explore the platform
Konnect
Build APIs
Kong Insomnia API Design API Testing and Debugging
Run APIs
API Gateway AI Gateway Event Gateway Service Mesh Kong Ingress Controller Kubernetes Operator
Govern APIs
Observability Metering & Billing Kong Identity Scorecards API Security & Access Control
Discover APIs
Developer Portal Catalog
Tools



We Value Your Privacy
English
By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.
(Privacy Policy)
Accept All
Customize Choices
Ask AI



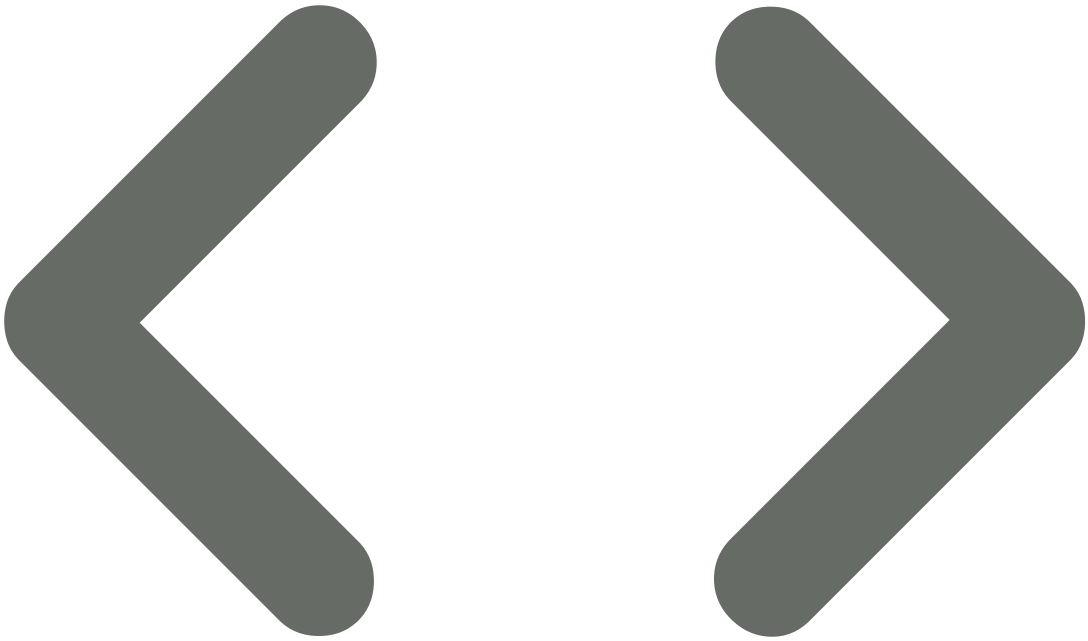
- kongctl

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



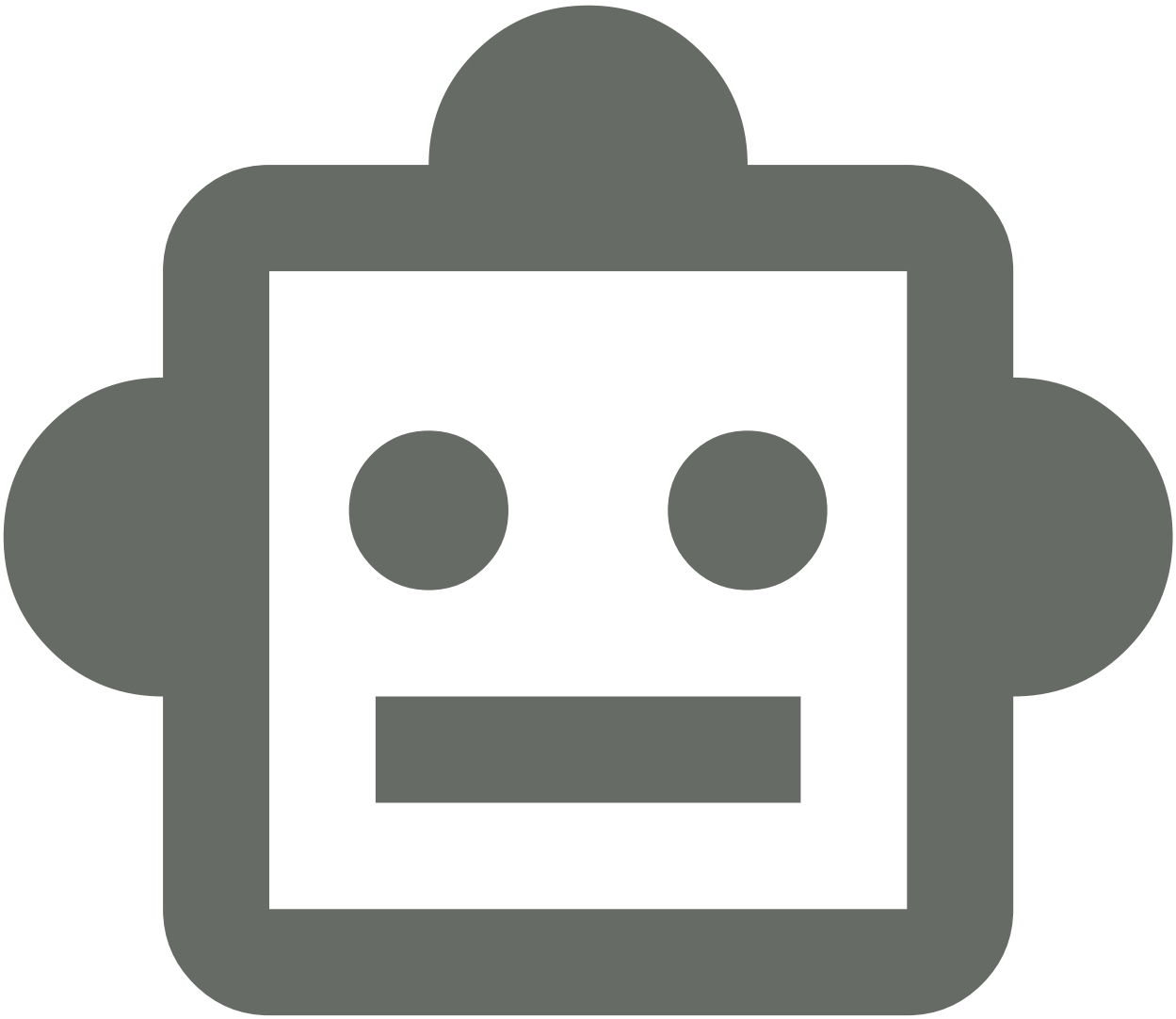
- Inso CLI

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



- Koh CLI

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



- Terraform

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



- Ingress controller
- View all →

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



Why Kong?

lify managing and securing APIs.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

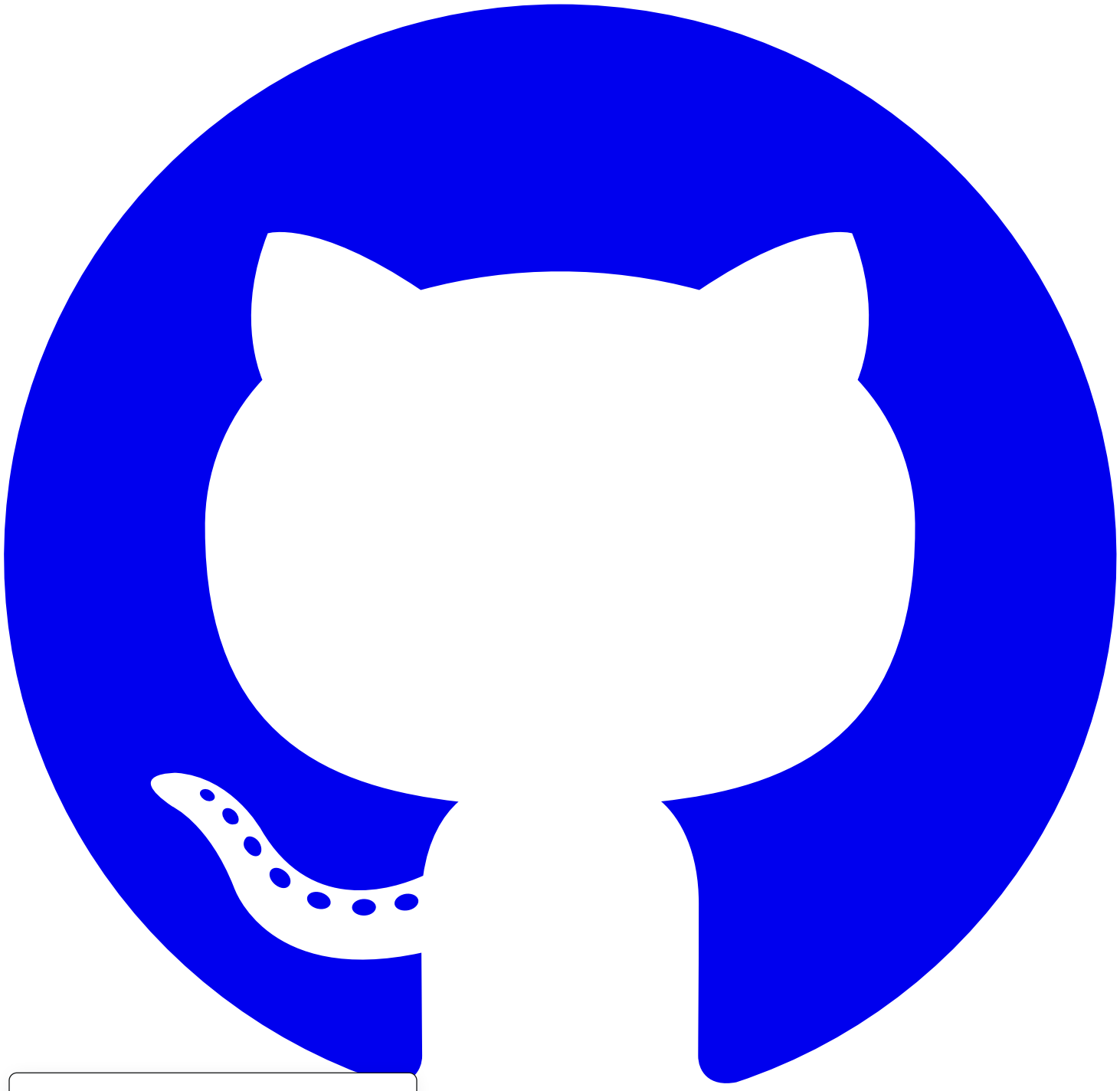
ecK kongctl Inso CLI Koh CLI

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

For product versions that have reached the end of sunset support, see the changelog archives.

3.16.0.0

Release date 2026/09/15

Breaking Change

Core

- Changed the Common Expression Language (CEL) backend from `cel-rust` 0.11.6 to `cel-cpp` 0.15.0. This is a breaking change for conditional plugin execution: `cel-cpp` is spec-compliant where `cel-rust` was not, so a small number of expressions involving quote-escaped strings or missing map keys (`kong.ctx.shared`, `principal.metadata`) now behave differently. See 3.16.x breaking changes for details.

Feature

Plugin

- **kafka-consume, confluent-consume**:
  - The consumer now transparently decompresses `gzip`, `snappy`, and `lz4` record batches.
- **kafka-upstream, confluent, kafka-log**:
  - Added a new `compression_type` configuration option that lets the producer compress message batches before sending them to the Kafka broker. Supported codecs are `none` (default), `gzip`, `snappy`, and `lz4`. Compression applies only to the Kong-to-broker hop and is independent of any HTTP-level `Content-Encoding`.
- **kafka-upstream, kafka-log, confluent**:
  - Added a `payload_encoding` option (`avro_json` or `simple_json`) to the schema registry `value_schema` and `key_schema` configuration. When set to `simple_json`, the gateway accepts plain JSON request bodies for Avro schemas and resolves union/nullable branches against the schema, so clients no longer need to type-tag union fields (e.g. `{"int": 1}`, `{"null": null}`). Defaults to `avro_json` to preserve existing behavior.
- acl:
  - Added `allow_when`/`deny_when` fields that evaluate CEL boolean expressions against the request context to provide more expressive access-control policies.
- ai-proxy-advanced:
  - Added `cache_read_cost` and `cache_write_cost` to `model.options` so cached and cache-write prompt tokens can be priced per 1M tokens, alongside `input_cost` / `output_cost`.
  - Added filtering for empty Anthropic thinking blocks in request history.
  - Added support for reporting prompt cache token metrics when using native (non-OpenAI) request/response formats.
  - Added support for reporting prompt cache token metrics in OpenAI-compatible format.
  - Added support for per-cache-TTL cache-write pricing, context-window pricing factors, and service-tier pricing factors via `model.options` (`cache_write_cost_list`, `context_window_factor`, `service_tier_factor`). Anthropic (cache TTL + service tier), OpenAI, and Gemini service tiers are collected from upstream responses.
- confluent:
  - Added `producer_config_enabled` (default `false`) to opt into honoring the `producer_*` delivery settings (asynchronous mode, request acks/limits/retries), which the plugin previously ignored while always producing synchronously. The default preserves the existing synchronous behavior for current configurations.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

t, `call` nodes verify the server's TLS certificate against a trust store built solely from the referenced CA `_certificate` set. Ignored when a node's `ssl_verify` is disabled.

dentity principal with the `property` node using `kong.client.principal` and nested fields such as

enforces per-customer access policies (credit balance and feature entitlements) retrieved from an

mit usage-event logging to responses whose status code falls within the given ranges, for example `200-`

esponse which, when set to `true`, returns validation errors in the RFC 9457 `application/problem+json`

- opa:
  - Added WebSocket (`ws`/`wss`) protocol support. OPA is evaluated at handshake time before the connection is established.
- openid-connect:
  - Added actor token support to token exchange. Configure `token_exchange.request.actor_token` to include an actor token, read from a request header or a static token value, in the token exchange request.
  - Added `protected_resource_metadata` configuration to support RFC 9728 OAuth 2.0 Protected Resource Metadata, enabling automatic authorization server discovery by serving a well-known metadata document and enriching `WWW-Authenticate` challenge headers with `resource_metadata` and `scope` attributes.
  - Added support for Microsoft Entra ID's On-Behalf-Of flow through a new `token_exchange.grant_type = jwt_bearer` (RFC 7523) mode, plus `map_identities_from` and `token_exchange.provider`.
  - `KONG_FIPS=on` no longer generates default EdDSA JWKS keys. Use an RSA-PSS or NIST-curve ECDSA key instead.

    ```
    If you are migrating from the FIPS package to the FIPS 3 package for compliance with FIPS 140-3, please note that this is a breakin
    ```

    Copied!
- opentelemetry:
  - Added a `metrics.enable_principal_attribute` config option. When enabled, the principal attributes are added to the metrics when an authenticated principal is available.
  - Added the log fields that plugins set with `kong.log.set_serialize_value` to the access log export, so users no longer need `custom_attributes_by_lua` to deliver them.
- proxy-cache-advanced:
  - Added `cache_by_principal` configuration option to switch cache key segmentation by Principal when using Kong Identity Directory. When enabled, the authenticated Principal's ID is used to compose the cache key.
- proxy-cache:
  - Added `cache_by_principal` configuration option to switch cache key segmentation by Principal when using Kong Identity Directory. When enabled, the authenticated Principal's id is used to compose the cache key.
- rate-limiting-advanced:
  - Added `expressions.limit` to drive individual limit entries via CEL, falling back to the static value on error.
  - Added support for rate limiting by Kong Identity Directory Principal, via `identifier: principal`, and for overriding the rate-limiting key entirely via the new, expressible `custom_key` field.
- rate-limiting:
  - Added a `custom_key` field for overriding the rate-limiting key entirely. The field is expressible, so it can be driven by a CEL expression, for example to rate limit by a Kong Identity Directory Principal's metadata.
  - Added support to rate limiting by Kong Identity Directory Principal.
  - The `second`, `minute`, `hour`, `day`, `month`, and `year` fields are now expressible, so the rate limit value itself can be driven by a CEL expression, for example to vary the limit by a Kong Identity Directory Principal's metadata.
- request-transformer-advanced:
  - Added a `max_request_body_size` configuration option that lets the plugin apply body transformations to request bodies that Nginx buffers to a temporary file (bodies larger than `client_body_buffer_size`). Previously such bodies were never read, so the configured body transformations were silently skipped and the original body was proxied upstream. The option defaults to `-1`, which preserves that behavior; set it to a positive byte limit to opt in (or `0` for unlimited, still bounded by Nginx's `client_max_body_size`).

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

fault `block`) that allows users to set the plugin to `log_only` mode, where validation failures are logged but

ent authentication method.

message as new optional value for `config.text_source`.

tion-health gating (`producer_async_health_gating`, default `false`): when the producer is sustainedly broker, asynchronous requests fail fast with HTTP 503 instead of returning 200 and silently dropping the n authenticate and reach the broker again. Transient produce errors keep async best-effort buffering.

confluent-consume: Added `schema_registry.confluent.authentication.logical_cluster_id` and rget-sr-cluster and `Confluent-Identity-Pool-Id` request headers when authenticating to Confluent

- Added a FIPS-compliant variant of the distroless image.
- Added an info-level log entry at startup that records the running Kong Gateway version.
- Added an optional queue-level circuit breaker (`failure_threshold`, `breaker_cooldown`; off by default) for all plugins using `config.queue` that sheds buffered entries when a destination is permanently down instead of retrying indefinitely.
- Added Debian 13 (Trixie) support for building packages.
- Added info-level logging for failed Kong Identity principal authentication and lookup attempts. Kong Gateway now logs the identity service's failure reason, for example an invalid password or a missing credential.
- Added support for driving certain plugin configuration fields with a CEL expression, evaluated per request and falling back to the field's static value if unset or if evaluation fails.
- Added the `analytics_enabled` configuration option, allowing operators to disable the Konnect analytics feature while keeping `konnect_mode` enabled. It defaults to `on`, preserving existing behavior.
- Added the `plugin_ordering_algorithm` configuration option. It selects how Kong Gateway orders plugins that declare a dynamic `ordering` rule. The default, `legacy`, keeps prior behavior, so upgrading does not change plugin execution order. Set the option to `priority_preserving` to move only the plugins that declare an `ordering` rule; every other plugin then keeps its configured priority order. This is a node-level setting. In hybrid mode, set it on each data plane and restart the data plane. An `ordering` rule that names an unknown plugin, or names the plugin itself, now logs a warning each time the configuration changes. It previously logged an info message on every request.
- Allowed DB-less nodes to change their log level through the debug server (`debug_listen`/`debug_listen_local`) with `kong debug log_level set`. The Admin API equivalent stays unavailable in DB-less because it cannot be protected by RBAC.
- Enabled TLS 1.3 for the FIPS cipher suite in the FIPS-140-3 artifacts, and configured those builds to use AES-based TLS 1.3 ciphersuites. The FIPS-140-2 artifacts keep TLS 1.2.
- **Hashicorp Vault**: The AWS IAM credential fields `aws_access_key_id`, `aws_secret_access_key`, and `aws_assume_role_arn` are now referenceable, so you can store them as vault references instead of literal values.
- Kong now publishes one artifact per FIPS certification level. The `kong-enterprise-edition-fips` package and the `<version>-<distro>-fips` image keep the OpenSSL FIPS provider 3.0.9 (FIPS 140-2). The new `kong-enterprise-edition-fips-140-3` package and the `<version>-<distro>-fips-140-3` image use the provider 3.1.2 (FIPS 140-3).
- **Konnect Only**. Added support for annotating custom lua code frames in CPU Profiling.
- **Konnect Only**. **Analytics**: Added the OpenID Connect plugin's `credential_claim` identity to Analytics events, so requests are attributed under the auth dimension in Konnect without a Kong Consumer.
- **Plugin conditions**: Plugin condition expressions are now type-checked at configuration time. Expressions whose result is provably non-boolean (for example `1 + 1`, `"foo"`, or `size(x)`) are rejected when the plugin is configured, instead of failing at request time. Boolean and statically-unknown expressions are accepted as before.
- The FIPS-140-3 artifacts serve an explicit TLS 1.2 cipher list: four ECDHE suites with AES-GCM. The FIPS-140-2 artifacts keep the OpenSSL FIPS alias, which also admits static RSA key transport and SSLv3-era suites. To serve a different list, set `ssl_cipher_suite = custom` and `ssl_ciphers`.

CLI Command

- Added a `kong workspace` CLI command. `kong workspace rename OLD_NAME NEW_NAME` renames a workspace (cascading the new name to RBAC role-endpoint permissions), and `kong workspace detect-admin-endpoint-conflicts` lists workspaces whose names collide with admin API URL prefixes.

PDK

- Added a new Metrics PDK (`kong.metrics`) that enables custom plugins to define and record metrics, with initial support integrated into the `opentelemetry` plugin.
- Added the `kong.client.get_principal()` PDK function. It returns the Kong Identity principal authenticated for the current request, similar to `kong.client.get_consumer()`.

The field reports the `id` and `display_name` of the authenticated principal for the request.

C

C

onnections. Set `cloud_authentication.auth_provider` to `oauth` on any Redis configuration (Rate
nnect, and other Redis-backed plugins) to authenticate using a bearer token from an OAuth 2.0 token

K

orm renderer.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- **AI Gateway**:
  - Added Gemini and Vertex mapping for the `googleSearch` tool grounding metadata chunks, which are now mapped into the `annotations` section of the OpenAI-format chat response.
- **AI plugins (Redis vector database)**:
  - Fixed an issue where a schema-compatible Redis vector index could be spuriously dropped and recreated — emptying it — causing transient empty results from semantic cache, RAG, and semantic-guard lookups. The `FT.INFO` index-compatibility check now reads index attributes by key name instead of by position, so a compatible index is no longer misjudged and dropped.
- **Konnect Only**. **datakit**:
  - Added node source and runtime phase metadata to Datakit tracing events so Konnect Debugger can reconstruct the Datakit node graph from captured trace data.
  - Fixed an issue where several datakit instances (the base plugin plus clones) that ran in one request overwrote each other's active-tracing capture. Each datakit instance now reports its own capture, keyed by its plugin id, and the capture payload carries the `plugin_name` field. Datakit plugin spans also carry a `proxy.kong.plugin.ref` attribute set to the base plugin name.
- **ai guardrail plugins**:
  - Fixed an issue where a JSON `null` field in a response from the third-party service made the plugin return `500`.
  - Fixed issues where the AI Sanitizer and AI AWS Guardrail plugins couldn't work properly with Claude Code.
- acl:
  - Fixed an issue where the ACL plugin returned 403 for authenticated credentials with no consumer and no groups (e.g. ACE / Dev Portal application credentials) when configured with a deny list. A group-less identity cannot match a deny list and should be allowed through.
  - Fixed an issue where the ACL plugin ignored a consumer's ACL groups when `include_consumer_groups` was enabled and the consumer belonged to an enterprise consumer group, causing incorrect `403` denials (with `allow`) or incorrect access (with `deny`) for authenticated and anonymous consumers. The plugin now falls back to the consumer's ACL groups when no third-party authenticated groups are present.
- acme:
  - Fixed an issue where an expired `fail_backoff` key would keep blocking certificate issuance. Retries are now rejected only while the backoff timestamp is still in the future.
- ai-aws-guardrails:
  - Fixed an issue where the plugin ignored the configured `text_source` option when `allow_masking` was enabled, masking the entire raw request body instead.
  - Fixed an issue where the request to the AWS Guardrails service could return a 500 error due to a failure in request signature.
- ai-mcp-oauth2:
  - Fixed a JWT security vulnerability.
- ai-mcp-proxy:
  - Fixed an issue where the ACL decision cache has collision.
  - Fixed an issue where non-managed MCP listeners advertised unsupported SSE streams.
  - Fixed an issue where converted HTTPS MCP tool calls lost the original forwarded port on the internal TLS Unix socket.
  - Fixed an issue where gzip-compressed and malformed MCP request bodies were incorrectly rejected with HTTP 413.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

quests sent in a provider's native format instead of the OpenAI-compatible format.

l fail to decode when the client's request advertised Brotli support, since Anthropic's edge may then return pression – which only understands gzip – can't inflate. Kong now normalizes the outgoing `Accept-` LM provider: `gzip, identity` when the client accepts gzip, and `identity` otherwise, so a client that dy on the paths that pass the upstream response through untouched.

usage metadata was lost in streaming responses.

limits were discarded or overridden during request normalization.

s did not correctly handle empty 'requires' array.

ens and `cache_read_input_tokens` were excluded from prompt token counts, causing undercounting

led to Databricks upstreams, causing a 400 error since Databricks does not support this field.

- Fixed an issue where native Anthropic streaming counted provisional `message_start` output tokens and summed cumulative `message_delta` usage.

- Fixed an issue where cache_control in Bedrock Anthropic format requests using ARN model IDs was lost.
- Fixed an issue where Bedrock native ConverseStream requests using ARN model IDs could fail while handling streamed responses.
- Fixed an issue where the `context-management-2025-06-27` Anthropic beta flag was dropped when routing to a Bedrock target, causing Bedrock to reject requests carrying `context_management` with a `400` error on failover from an Anthropic target.
- Fixed an issue where output token usage was reported as zero for Bedrock native format request to AWS models like Nova.
- Fixed an issue where signed Anthropic thinking blocks with omitted text were removed from request history.
- Fixed an issue where Anthropic-format requests could report zero usage when it targets Bedrock's own model via conversion.
- Fixed an issue where full-sync reconfigure reset balancer state even when plugin configs were unchanged.
- Fixed an issue where OpenAI, Azure, and Gemini realtime upstream URLs using `wss` were downgraded to `ws`.
- Fixed an issue where a credential header could appear unredacted in access logs when using AWS temporary credentials with Bedrock.
- Fixed an issue where Ollama rejected OpenAI message content arrays.
- Fixed an issue where OpenAI content array was sent as bedrock/gemini system message which only support string.
- Fixed an issue where the configured auth header (e.g. `api-key`, `x-api-key`) leaked the provider credential into access logs instead of being redacted.
- Fixed an issue where token usage for Anthropic requests using the compaction beta was undercounted. Kong now sums `usage.iterations` when present instead of relying solely on the top-level `input_tokens` / `output_tokens`, which only reflect non-compaction iterations.
- Fixed an issue where a malformed RSA private key in a Gemini provider's `service_account_json` caused crash with SIGSEGV
- Fixed an issue where adapted Anthropic responses omitted cache and reasoning usage details.
- Fixed an issue where Claude Sonnet 5 and Opus 5 rejected OpenAI-format reasoning requests on Bedrock due to an unsupported thinking.type value.
- Fixed an issue where repeated cumulative usage in streaming responses inflated token counts.
- Fixed an issue where Gemini non-streaming responses that reached `max_tokens` reported a `stop` finish reason instead of `length`.
- Fixed an issue where a Gemini response with JSON `null` token counts in `usageMetadata` caused an HTTP 500 on non-streaming chat requests and an empty HTTP 200 response on streaming chat requests.
- Fixed an issue where provider-reported reasoning-token details were not correctly reported.
- Fixed an issue where xAI reasoning tokens were not folded into completion_tokens, breaking the OpenAI usage identity.
- Fixed an issue where Gemini's tool-use token usage (billed when server-side tools such as grounding, code execution, or URL context are used) was not counted in the reported token usage or cost.
- Fixed Azure OpenAI responses support in deployment style.
- Fixed Gemini image generation token accounting to include reasoning tokens (`thoughtsTokenCount`) in `output_tokens` and report them as `output_tokens_details.reasoning_tokens`. Previously the reasoning tokens were dropped, which under-reported output tokens, analytics, and calculated cost for thinking-enabled image models.
- Fixed Gemini realtime (Live API) token accounting to include reasoning tokens (`thoughtsTokenCount`) in the reported output tokens, and to report them as `output_tokens_reasoning_tokens_count`. Previously the reasoning tokens were dropped, which under-reported the output and total token counts of thinking-enabled realtime sessions.
- Fixed multipart video generation requests to OpenAI-compatible providers leaking internal multipart metadata into JSON upstream bodies, which caused the requests to fail.
- Fixed token usage statistics being silently dropped for `audio/v1/audio/transcriptions` requests. Token-billed OpenAI/Azure transcription models (e.g. `gpt-4o-transcribe`) now report `input_tokens`/`output_tokens`/`total_tokens` and the `text_tokens`/`audio_tokens` breakdown in the analytics log when

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

oken was cached indefinitely and never refreshed, causing requests to fail with 401 after the token expired

g tokens in `completion_tokens` for both OpenAI-format and native-format proxying.

mazon MemoryDB vector store failed with `unexpected search response: no value found in result` RCH fields by name instead of by position, treats the vector distance as optional, and requests as many rows

stom_id matcher resolved to consumer username.

esponses route was not recognized.

  - Fixed an issue where Cohere native chat requests and responses were not extracted, so the plugin now detects and redacts PII in Cohere-format traffic.
  - Fixed an issue where AI Sanitizer returned a 500 status code when the provider was using another model, such as a `claude-*` model served by the `gemini` provider on Vertex AI.
- datakit:
  - Fixed an issue where replaced response bodies kept stale upstream validators and representation metadata.
  - Fixed an issue where the `response` node set `Content-Type: application/octet-stream` for `json_to_xml` output. It now defaults to `application/xml` when no explicit `Content-Type` is provided.
  - Fixed an issue where the `exit` node sent a string body without a correct `Content-Type`. It now uses the content type that the source node declares, for example `application/xml` for `json_to_xml` output, and defaults to `application/octet-stream` in the request phase.
  - Fixed CVE-2026-14916 by binding the JWT algorithm to the verification key's type in the `jwt_verify` node, so a symmetric algorithm can no longer be verified against an asymmetric key, preventing an authentication bypass where an attacker forges tokens by using the public key as an HMAC shared secret, under configurations where the node uses a PEM key string and `allowed_algorithms` is unset (any algorithm accepted).
  - Fixed the body handling of the `exit` and `service_request` nodes when the body input is a JSON null. The nodes sent the internal Lua value as the string "userdata: NULL". The `exit` node now sends no body. The `service_request` node now keeps the upstream request body.
- grpc-gateway:
  - Fixed an issue where the plugin mapped the gRPC `UNIMPLEMENTED` status to HTTP 500 instead of 501. The new option `use_standard_grpc_status_codes` (default `off`) enables the fix.
  - Fixed an issue where a single query-parameter value for a repeated integer or enum field failed to transcode, so such values are now correctly mapped to the gRPC request.
- ip-restriction:
  - Fixed an issue where warn log messages did not include the plugin name, making it difficult to identify the source of blocked requests.
- jwt-signer:
  - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.
- jwt:
  - Fixed an issue where the plugin failed to decode standard Base64 secrets that contained `+` or `/` when `config.secret_is_base64` was enabled.
- key-auth:
  - Fixed an issue where an invalid API key was not falling back to the configured anonymous consumer when `principals.error_on_miss` was set to `false`, unlike every other authentication failure.
- mocking:
  - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oas-validation:
  - Added a warning when structured_errors is enabled without verbose_response, as both options must be enabled together.
  - Fixed an issue where readOnly and writeOnly properties in OpenAPI 3.1.x specifications were not correctly validated.
- oauth2-introspection:
  - Fixed an authentication bypass where a revoked or expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false` result is now also cached briefly (5 seconds).

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

reak the WebSocket handshake.

udience conditions matched only the claims `scope` and `aud`. Support for `scopes_claim`/`audience_claim` is

t failed an authorization check (such as `scopes_required`) was not attributed to its consumer, consumer tics.

om the introspection endpoint was cached using the default/positive TTL instead of `cache_ttl_neg`, ger than configured.

was sent to the token endpoint for all grant types. It is now included only for the `authorization_code`

h `expires_in <= 0` was cached indefinitely with the proactive expiry re-check disabled, causing the to the upstream until a worker/process restart.

- Fixed an issue where `token_endpoint_auth_method`, custom token headers, and extra token POST args were ignored during token exchange, causing the exchange request to always use `client_secret_basic` regardless of configuration.
- Fixed an issue where `token_exchange.cache` defaulted to disabled instead of its documented enabled default when omitted.
- Fixed an issue where a session was not saved on a request that performed a token exchange. A session-restored request now reuses the exchanged identity instead of re-exchanging.
- Fixed an issue where disabling `verify_signature` gave no warning, even though it also skips signature verification for bearer tokens from clients. The plugin now logs a warning when `verify_signature` is `false` and `auth_methods` includes `bearer`. It logs a notice instead when `bearer` is not enabled. `token_exchange.subject_token_issuers` entries for a different issuer now also warn when they don't set `verify_signature` to `true`.
- Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.
- Fixed an issue where consumer groups could not be resolved when the groups claim was only available from the user info endpoint, which resulted in a `500` error.
- Fixed the token exchange cache to stop outliving its subject token, mixing entries across `grant_type`/`provider`/client, and serving stale results after a plugin update under incremental config sync. Use `cache_tokens_salt` to scope the cache further.
- The `ignore_signature` option now logs a warning when it is set, because it turns off JWS signature verification for the listed grants.

- opentelemetry:
  - Fixed `gen_ai.client.token.usage` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
  - Fixed an issue where a `custom_attributes_by_lua` key that conflicts with an existing log field stopped the log handlers of the plugins that run later.
  - Fixed an issue where metrics were rarely (or never) exported on data planes whose configuration is rebuilt more frequently than the metrics `push_interval`. The periodic metrics push timer is no longer cancelled and recreated on every configuration rebuild, so its `push_interval` countdown is preserved.
  - Fixed an issue where OpenTelemetry metrics only supported integer values by adding support for floating-point values in the OTLP exporter.
  - Fixed an issue where the `kong.dp.cluster_cert.expiry` metric reused the same datapoint timestamp across exports, causing monitoring systems to treat the series as stale.
  - Reduced per-request overhead of `resource_attributes` by skipping template rendering entirely for static (non-templated) fields.
- prometheus:
  - Fixed `kong_ai_llm_tokens_total` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
  - A histogram's series (its buckets, `_count` and `_sum`) is now read as one atomic group during a scrape. Previously the exporter yielded and flushed output between individual samples, so an observation landing mid-scrape could bump `_count`/`_sum` after the buckets were already read, producing an internally inconsistent histogram in a single response.
  - Fixed an issue where `kong_nginx_requests_total` and `kong_nginx_connections_total` duplicated the `http` values under the `stream` subsystem. They are now exported once under `http`.
  - Histogram series are now emitted grouped by label set (all buckets in ascending `le` order, then `_count`, then `_sum`) so the output follows the OpenMetrics interleaving rule. Previously samples were grouped by suffix, which caused strict consumers such as Google Managed Prometheus to drop points.
  - Histograms now expose every bucket (including those with a value of `0`) starting from the first observation of a given label set. Previously only buckets greater than or equal to the observed value were created, so lower buckets appeared lazily on later scrapes.
- proxy-cache-advanced:
  - Fixed an issue where the cache key did not correctly segment by consumer group.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

y segment by consumer group.

`body.custom` or `upstream.body.custom` maps caused the plugin to send `{}` request bodies.

ucceeded on retry still returned the earlier failed response. The plugin now returns the response from the

n `header:value` format (e.g. `correlationId:1222`) failed to remove the header. The value part is now avior from 3.11.

empted to change the status during body filtering.

  - Fixed an unreachable code path that prevented the plugin from falling back to validating the `samlp:Response`-level signature when the `saml:Assertion` was not individually signed. With `validate_assertion_signature` enabled, an unsigned assertion inside a signed Response is now accepted (and a Response with no valid signature on either the assertion or the root is rejected), so IdPs that sign the Response envelope rather than the assertion no longer require signature validation to be turned off. Relates to the Response-level fallback described in CVE-2026-14917.
  - Fixed CVE-2026-14917 by validating the `samlp:Response` signature whenever an `idp_certificate` is configured – even when `validate_assertion_signature` is disabled – and logging a warning when neither is set, preventing an unauthenticated attacker from impersonating any user with a crafted unsigned SAML response, under configurations that set `validate_assertion_signature` to `false`.
  - Hardened signature validation against XML Signature Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that names the `ID` of the element it digests and consumes, so a signature can no longer be bound to a different element than the one that provides the identity. A response whose reference URI is empty or not a same-document fragment is now rejected.

- session:
  - Fixed an issue where when Postgres strategy is used for session storage, concurrent authentication requests could result in race condition and session creation collisions.
- tcp-log:
  - Fixed the race condition where the balancer `tries` array gets cleared before the asynchronous timer thread can safely encode it.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins' request extractors did not correctly honor the configured `text_source` option.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins failed to extract prompt text from OpenAI-native requests using the `input_text` content type.
- ai-proxy, **ai-proxy-advanced**: Fixed an issue where Gemini streaming analytics could over-report token usage. Gemini repeats its usage metadata on every stream chunk as a running request-level total, but the counts were summed across chunks instead of replaced, multiplying the reported usage by the number of chunks. Reasoning tokens were affected the most, because Gemini reports them in full on the first chunk and does not change them afterwards. A chunk that carried no usage metadata also reset the recorded output token count to zero.
- ai-proxy and **ai-proxy-advanced**: Fixed a 500 crash on Bedrock Converse requests when a `cachePoint` content block immediately followed a `toolResult` block in the same message.
- ai-semantic-prompt-guard, **ai-semantic-response-guard**, **ai-proxy-advanced**, **ai-semantic-cache**: Fixed vector database indexes and cached embedding records being left behind when a plugin instance was deleted.
- kafka-consume, **confluent-consume**: Fixed an issue where the consumer group subscription declared protocol version 1 but carried the version 0 layout, so a topic-rewriting proxy rejected it.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed the confluent schema registry client caching fetched schemas and subjects under keys that did not account for the registry URL or authentication. Instances configured with different credentials could be served each other's cached lookups, letting an unauthorized instance bypass schema-registry authentication once any authorized instance had warmed the cache. Cache entries are now scoped to the registry identity, and the configured `schema_registry.confluent.ttl` is now applied to these cache entries.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a crash that occurred when a plugin set `schema_registry.confluent.authentication.mode` to `oauth2` but omitted the `oauth2_client` record. Previously this returned a 500 from the Admin API during validation, and once saved it also crashed at runtime while fetching the schema registry OAuth2 token. The absent `oauth2_client` is now handled and falls back to the default authentication method, so a configuration without `oauth2_client` works end-to-end.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a permanent failure that started when a broker answered an `ApiVersions` request with an error code, an empty API key set, or a malformed frame. The client cached that result and every later request failed with `ProduceRequest API version not negotiated; broker may be unreachable` until the node restarted, even after the broker recovered. The client now refuses an unusable `ApiVersions` response, keeps the previous negotiation, and retries the negotiation for a topic whose metadata is already cached.

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

2533.

6434.

0005.

2055 and CVE-2026-48142.

in-API driven updates) under the default `worker_consistency = eventual`. When an upstream and its get event could run before the deferred balancer creation, log `balancer not found` and drop the target, facing as 503 / DNS name errors). The target event now creates the balancer itself when it is missing.

ces could match either `/routes/:name` or `/:workspace_name/services`. Now non workspace-specific e-specific routes like `/:workspace_name/services`, ensuring correct route matching for Admin API

ed the balancer rebuild when the worker-local target cache was empty and the event carried a partial target

per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.

ng insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.

- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.
- Fixed an issue where a request context in the stream TLS proxying path was not returned to the internal context pool, which could reduce the pool's efficiency under sustained TLS stream traffic.
- Fixed an issue where an in-flight request that completed after its upstream target became unavailable could restore the target to the latency balancer and route subsequent requests to it.
- Fixed an issue where cluster mutex did not timeout immediately with `no_wait = true` when node level mutex is being held by another worker.
- Fixed an issue where custom Lua running with `untrusted_lua` set to `sandbox` could no longer create global variables or modify allowed library tables.
- Fixed an issue where custom streamed plugin schemas were still sandboxed when `untrusted_lua` was set to `on`, so schema code that needed to opt out of sandboxing (unlike custom plugin handlers, which already respected `on`) was incorrectly rejected on both the control plane and the data plane.
- Fixed an issue where dynamic plugin ordering (`ordering`) leaked into the downstream phases (`header_filter`, `body_filter`, `response`, `log`). Dynamic ordering now again applies only to the `access` phase; downstream phases execute in static plugin-priority order.
- Fixed an issue where immediate cosocket SSL handshake errors could be reported as subsequent socket write failures.
- Fixed an issue where Kong Gateway could crash while closing a WebSocket connection.
- Fixed an issue where Kong Gateway crashed in the stream preread phase when the upstream could not be resolved, due to calling an HTTP-only API in the L4 proxy path.
- Fixed an issue where Kong Identity principal caching ignored the gateway's `db_cache_ttl` and `db_cache_neg_ttl` settings. The gateway now uses the lower of the TTL returned by the identity service and its own configured TTL. If neither side provides a usable TTL, the gateway falls back to a bounded internal default instead of caching the entry forever.
- Fixed an issue where Kong skipped the WebSocket `ws_close` phase when it could not proxy the request.
- Fixed an issue where memory tracing didn't generate any events.
- Fixed an issue where response buffering was forced on routes with a plugin that has a response handler, breaking streaming responses even when the plugin did not read the response. Response buffering is now decided after the access phase, so plugins can no longer read the final value of `ngx.ctx.buffered_proxying` in `access` phase.
- Fixed an issue where `/status/ready` could report a data plane ready before the new router and plugins iterator became active. Kong now reports readiness only after plugin configuration completes and the live objects are installed.
- Fixed an issue where streamed custom plugins were not recognized when configured in the `ordering` field of `plugins` on Data Planes.
- Fixed an issue where the error wrapper was not being freed after sandboxed template rendering in plugins such as `request-transformer` and `response-transformer`.
- Fixed an issue where the priority field of a cloned plugin could not be updated while it was still referenced by plugin instances.
- Fixed an issue where upstream failover could return intermittent 503 responses while a failover target remained healthy.
- Hardened HashiCorp Vault cloud authentication (AWS, Azure, GCP): Kong keeps credentials out of internal cache keys, truncates Vault error responses in logs, sets timeouts on all authentication requests, and applies `ssl_verify` to every outbound TLS call.
- **Keyring**: Fixed a Lua error in the keyring cluster strategy's `request_missing_keys` timer callback that crashed (`attempt to index a nil value`) when the database became unavailable, because the `keyring_meta:each()` iterator yields a nil row alongside the error. The error is now logged and the row skipped.
- **Konnect Only**. **Debugger**: Added new debugger span attributes `tls.client.server_name`, `proxy.kong.upstream.address`, `proxy.kong.request.host`, `http.request.header.connection`, `proxy.kong.client.connection.request_count`. Removed debugger span attributes `proxy.kong.upstream.host`, `proxy.kong.upstream.addr`, `http.request.header.host`, `proxy.kong.client.keepalive`
- **Konnect Only**. **Debugger**: Fixed an error that could occur under sustained HTTPS traffic.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

that ended the request early with `kong.response.exit` during the response phase (for example a Datakit
s unfinished in the exported trace.

enting third-party HTTP client calls produced two indistinguishable `kong.io.http.request` spans for
`kong.io.http.request_uri` span, and the `connect`, DNS, and internal `request` spans it triggers are
also reports the `http.url` attribute.

py of the worker logs stack overflowed by replacing `deep_copy` with `cycle_aware_deep_copy`.

onse body of the external HTTP call `httpc:request()` was not captured.

y to instrument to the `ready_body()` and `body_reader()` functions to capture the response body.
ibute to the new `kong.io.http.read_resp_body` span and the `kong.io.http.request` span.
`_resp_body` span to the associated `kong.io.http.request` span.
ptured request/response headers/body items.
s (e.g. network errors, timeouts, etc.) to avoid capturing corrupted data.
oving the creation time of spans.
onse headers from `kong.io.http.request_uri` span to the `kong.io.http.request` span.

ity principal lookup was cached for up to 15 minutes. It now uses the feature's shorter default TTL instead.

nd start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with a clear error on an unterminated quote or `${`, instead of passing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

- **PostgreSQL OAuth auth**: Refined the token cache key to be derived only from the fields that identify an OAuth configuration.
- **PostgreSQL OAuth auth**: The OAuth discovery/token requests issued during the `init` phase now trust the CAs configured via `lua_ssl_trusted_certificate`. Previously this init-phase path used LuaSocket/LuaSec, which only honors the OS trust store, so an HTTPS identity provider using a private CA failed TLS verification during `kong start` even when its CA was configured.

Clustering

- **Clustering**: Fixed an issue where slow response handling in DP could prevent a recovery connection with CP.
- **Clustering**: Improved RPC connection stability during long-running calls: a data plane now cancels a timed-out RPC so a late response cannot disrupt later calls, and the connection is kept alive (within a bound) while calls are still in flight instead of being dropped as a missed heartbeat.
- **Clustering**: The `/status/sync` endpoint now returns an empty stat object instead of null before the first sync has run, keeping the response shape consistent.
- **datakit**: When a data plane older than the control plane does not support a datakit configuration, the control plane now removes only the incompatible datakit plugin instead of rejecting the whole configuration. This matches the Konnect control plane version-compatibility behavior.
- Fixed an issue where data planes retried unrecoverable TLS certificate setup failures instead of loading their backup configuration.
- Fixed an issue where heavy packet loss may prevent the full sync from resuming.
- Fixed an issue where incremental config sync failed when an entity referenced a non-exported entity, for example rbac_users.
- Fixed an issue where incremental sync delta coalescing merged non-adjacent deltas of the same entity, which could drop an entity's earlier create and keep only a later update, reordering it after a delta that already referenced it (e.g. a plugin instance referencing a cloned plugin by name) and causing validation to fail with "plugin '' not enabled". Coalescing now only merges successive deltas of the same entity, preserving the causal order the data plane relies on.
- Fixed an issue where pushing a configuration with a datakit plugin whose node `inputs` contained a null value to an older data plane could crash the control plane's config compatibility checker with "attempt to index local 'value' (a userdata value)".
- Fixed an issue where self-managed control planes rejected updates to a streamed plugin's handler while the plugin was referenced by a plugin instance.
- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.
- Fixed an issue where updates to a streamed plugin's definition (e.g. `handler`), or to a cloned plugin's definition (e.g. `priority`), were not applied on data planes when no referencing plugin instances changed. The data plane now rebuilds its plugins iterator so the updated definition takes effect, instead of continuing to execute the previous version.
- Fixed an issue where, with plugin streaming enabled, a single incremental sync validation failure would wipe out all plugin subschemas because the currently loaded plugins were not correctly excluded from the subschema unload.

PDK

- Fixed a race condition issue where the `tries` field of log serialization was not copied, causing data mutations when logs were generated concurrently via async timers.
- Fixed an issue on HTTP/2 connections where the request method and query were reported incorrectly, or could raise an error, when a request hit the error handler such as a 502 from an upstream connection reset.
- Fixed an issue where dynamic upstream TLS settings set via the PDK were lost during buffered proxying.
- Fixed an issue where `kong.response.set_header()` and `kong.response.clear_header()` could not replace or clear upstream `WWW-Authenticate` headers.
- Fixed an issue where the HTTP-to-gRPC status mappings for 400, 404, 429, 502, and 504 were incorrect. The new configuration option `use_standard_grpc_status_codes` (default `off`) enables the fix.

Admin API

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

or developer failed with "Old password is invalid" when the credential was created before the basic-auth
(e.g. a `kong_admin` user created in an older version and carried over through an upgrade). The old
erification logic that supports both SHA1 and SHA256 hashed passwords.

for HTTP/2 POST, PUT, and PATCH requests that omitted both Content-Type and Content-Length.

uld return an internal server error instead of a clear 400 response when given malformed or missing

On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading
ful exit no longer blocks a restart.

`_ssl_verify_depth` from 1 to 5.

Dependency

Default

- Added RHEL 10 (el10) to build.
- Added Ubuntu 26.04 (Resolute Raccoon) to build.
- Bundled the lz4 compression library (built from source, installed to `/usr/local/kong/lib`) so the Kafka plugins' `lz4` codec no longer depends on the base image incidentally providing `liblz4`.

Core

- Added RHEL 10 FIPS packages.
- Added the OpenSSL FIPS provider 3.1.2 for the FIPS-140-3 artifacts. The FIPS-140-2 artifacts keep the provider 3.0.9.
- Added Ubuntu 26.04 (Resolute Raccoon) FIPS packages and image.
- Bumped avro-c to 1.12.2.
- Bumped jansson to 2.15.1.
- Bumped jq from 1.8.1 to 1.8.2.
- Bumped kong-pgmoon from 2.3.4.0 to 2.3.5.0, which fixes SCRAM channel binding failing to connect to a PostgreSQL server whose certificate is signed with SHA-512.
- Bumped kong-redis-cluster from 1.6.0 to 1.6.1. This accepts the `x.y` Redis version format reported by some managed services (e.g. AWS ElastiCache Serverless Valkey) by making the patch component optional, and only warns on Redis version `9.0` or newer instead of `8.0` or newer.
- Bumped libexpat to 2.8.4.
- Bumped libxml2 to 2.15.4.
- Bumped lua-kong-nginx-module from 0.20.0 to 0.21.0.
- Bumped lua-resty-cel from 0.2.0 to 0.5.0.
- Bumped lua-resty-lmdb from 1.6.1 to 1.6.2 (includes LMDB library bump from 0.9.33 to 0.9.35).
- Bumped lua-resty-openssl to 1.9.0.
- Bumped lua-resty-simdjson from 1.2.0 to 1.3.0 (updates simdjson from 3.12.0 to 4.6.4).
- Bumped msgpack-c to 7.0.2.
- Bumped OpenSSL from 3.5.7 to 3.5.8.
- Bumped PCRE2 to 10.48.
- Bumped Solace from 0.1.0 to 0.2.0 (updates Solace client from 7.33.0.3 to 7.33.2.3).
- Now uses `cel-cpp` `0.15.0` instead of `cel-rust` `0.11.6` for Common Expression Language backend.
- Removed the legacy libpcre3 package dependency from Kong Gateway .deb packages since Kong Gateway already bundles PCRE2.
- Removed the legacy pcre package dependency from Kong Gateway .rpm packages since Kong gateway already bundles PCRE2 and RHEL 10 no longer ships legacy PCRE package.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

a large configuration sync from the control plane, reducing the chance of a worker restart while the update

arge incremental config update.

incremental sync enabled under some circumstances.

r version compatibility. This reduces CPU usage and transient memory pressure.

es by notifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in

sync by updating service data in place instead of triggering a full router rebuild when only service attributes

- Improved schema performance through miscellaneous tweaks.
- Improved the performance of dynamic plugin ordering. When a workspace's `ordering` rules do not depend on the request – for example, when they are declared only on global plugins – Kong Gateway now computes the plugin order once and reuses it for later requests, instead of recomputing it on every request. Workspaces with route-, service-, or consumer-scoped `ordering` rules keep computing the order per request, because that order can depend on the request. This change does not affect plugin execution order.
- On SSL/TLS requests the `ngx.ctx` table is now properly retrieved from a pool. This prevents unnecessary table rehashes.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

Admin API

- Improved the performance of listing workspaces with `?counter=true`. Entity counts for the whole page are now retrieved with one grouped query per entity type instead of a separate count query per entity type per workspace, while keeping the counts exact.

Plugin

- oas-validation:
  - Improved plugin performance by removing unnecessary hash computation. Enabling the plugin now adds much less latency and CPU overhead to requests.
- opentelemetry:
  - Reduced memory usage of the access log queue by deferring OTLP payload conversion to export time.
- kafka-upstream, **kafka-log**, **confluent**: Added an improved asynchronous Kafka producer, opt-in via the new `new_kafka_async_producer` config option (default `false`). When enabled, it batches and sends messages more efficiently under high load. Messages without a key are packed into larger per-partition batches; because Kafka does not guarantee ordering across partitions, their relative order may change. Set a message key if ordering matters — keyed messages are unaffected. The legacy producer remains the default.

3.15.0.6

Release date 2026/09/16

Bugfix

Plugin

- **ai-guardrails**:
  - Fixed an issue where a JSON `null` field in a response from the third-party service made the plugin return `500`.
- ai-aws-guardrails:
  - Fixed a issue where the request to the AWS Guardrails service could return a 500 error due to a failure in request signing.
- ai-mcp-proxy:
  - Fixed an issue where converted HTTPS MCP tool calls lost the original forwarded port on the internal TLS Unix socket.
  - Fixed an issue where gzip-compressed and malformed MCP request bodies were incorrectly rejected with HTTP 413.
- ai-proxy-advanced:
  - Fixed an issue where a Gemini response with JSON `null` token counts in `usageMetadata` caused an HTTP 500 on non-streaming chat requests and an empty HTTP 200 response on streaming chat requests.

oken details were not correctly reported.

t folded when visible completion tokens exceeded reasoning tokens.

t folded into `completion_tokens`, breaking the OpenAI usage identity.

sage metadata was lost in streaming responses.

limits were discarded or overridden during request normalization.

am requests using ARN model IDs could fail while handling streamed responses.

lancer state even when plugin configs were unchanged.

sent as a Bedrock/Gemini system message, which only supports strings.

in a Gemini provider's `service_account_json` caused a crash with SIGSEGV.

omitted cache and reasoning usage details.

ejected OpenAI-format reasoning requests on Bedrock due to an unsupported `thinking.type` value.

streaming responses inflated token counts.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where Gemini non-streaming responses that reached `max_tokens` reported a `stop` finish reason instead of `length`.
  - Fixed an issue where `cache_control` in Bedrock Anthropic format requests using ARN model IDs was lost.
  - Fixed multipart video generation requests to OpenAI-compatible providers leaking internal multipart metadata into JSON upstream bodies, which caused the requests to fail.
- ai-rag-injector:
  - Fixed an issue where a chunk lookup against an Amazon MemoryDB vector store failed with `unexpected search response: no value found in result set`. The MemoryDB driver now reads the `FT.SEARCH` fields by name instead of by position, treats the vector distance as optional, and requests as many rows as `fetch_chunks_count` asks for.
- ai-rate-limiting-advanced:
  - Fixed an issue where policies using a consumer `custom_id` matcher resolved to the consumer username instead.
- ai-sanitizer:
  - Fixed an issue where AI Sanitizer returned a 500 status code when the provider was using another model, such as a `claude-*` model served by the `gemini` provider on Vertex AI.
  - Fixed an issue where Cohere native chat requests and responses were not extracted, so the plugin now detects and redacts PII in Cohere-format traffic.
- basic-auth:
  - Fixed an issue where invalid credentials were not falling back to the configured anonymous consumer when `principals.error_on_miss` was set to `false`, unlike every other authentication failure.
- datakit:
  - Fixed an issue where the `exit` node sent a string body without a correct `Content-Type`. It now uses the content type that the source node declares, for example `application/xml` for `json_to_xml` output, and defaults to `application/octet-stream` in the request phase.
  - Fixed the body handling of the `exit` and `service_request` nodes when the body input is a JSON null. The nodes sent the internal Lua value as the string `"userdata: NULL"`. The `exit` node now sends no body, and the `service_request` node now keeps the upstream request body.
- jwt:
  - Fixed an issue where the plugin failed to decode standard Base64 secrets that contained `+` or `/` when `config.secret_is_base64` was enabled.
- key-auth:
  - Fixed an issue where an invalid API key was not falling back to the configured anonymous consumer when `principals.error_on_miss` was set to `false`, unlike every other authentication failure.
- oas-validation:
  - Added a warning when `structured_errors` is enabled without `verbose_response`, as both options must be enabled together.
  - Fixed an issue where `readOnly` and `writeOnly` properties in OpenAPI 3.1.x specifications were not correctly validated.
  - Fixed an issue where OpenAPI 3.1.x parameter patterns using ECMA-262 `\uXXXX` Unicode escapes failed to match correctly, because PCRE does not support that escape syntax.
- openid-connect:
  - Fixed an issue where disabling `verify_signature` gave no warning, even though it also skips signature verification for bearer tokens from clients. The plugin now logs a warning when `verify_signature` is `false` and `auth_methods` includes `bearer`. It logs a notice instead when `bearer` is not enabled. `token_exchange.subject_token_issuers` entries for a different issuer now also warn when they don't set `verify_signature` to `true`.
- opentelemetry:

a key that conflicts with an existing log field stopped the log handlers of the plugins that run later.

exported on data planes whose configuration is rebuilt more frequently than the metrics `push_interval`. led and recreated on every configuration rebuild, so its `push_interval` countdown is preserved.

ucceeded on retry still returned the earlier failed response. The plugin now returns the response from the

edrock Converse requests when a `cachePoint` content block immediately followed a `toolResult` block in

ai-proxy-advanced, **ai-semantic-cache**: Fixed vector database indexes and cached embedding records

e the consumer group subscription declared protocol version 1 but carried the version 0 layout, so a topic-

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **datakit**: When a data plane older than the control plane does not support a Datakit configuration, the control plane now removes only the incompatible Datakit plugin instead of rejecting the whole configuration. This matches the Konnect control plane version-compatibility behavior.

Core

- Fixed an issue where Kong Identity principal caching ignored the gateway's `db_cache_ttl` and `db_cache_neg_ttl` settings. The gateway now uses the lower of the TTL returned by the identity service and its own configured TTL. If neither side provides a usable TTL, the gateway falls back to a bounded internal default instead of caching the entry forever.
- Fixed an issue where Kong skipped the WebSocket `ws_close` phase when it could not proxy the request.
- **Konnect Only**. **Debugger**: Added new debugger span attributes `tls.client.server_name`, `proxy.kong.upstream.address`, `proxy.kong.request.host`, `http.request.header.connection`, `proxy.kong.client.connection.request_count`. Removed debugger span attributes `proxy.kong.upstream.host`, `proxy.kong.upstream.addr`, `http.request.header.host`, `proxy.kong.client.keepalive`.
- **Konnect Only**. Fixed an issue where a failed Kong Identity principal lookup was cached for up to 15 minutes. It now uses the feature's shorter default TTL instead.

PDK

- Fixed an issue where `kong.response.set_header()` and `kong.response.clear_header()` could not replace or clear upstream `WWW-Authenticate` headers.

Dependency

Core

- Bumped avro-c to 1.12.2.
- Bumped jansson to 2.15.1.
- Bumped libexpat to 2.8.4.
- Bumped libxml2 to 2.15.4.
- Bumped lua-resty-openssl to 1.9.0.
- Bumped msgpack-c to 7.0.2.
- Bumped OpenSSL from 3.5.7 to 3.5.8.
- Bumped PCRE2 to 10.48.

Performance

Clustering

- **Clustering**: Improved DP performance when applying a large incremental config update.
- Improved configuration synchronization latency in some cases when incremental sync is enabled.

Core

- Improved performance when `untrusted_lua` is set to `sandbox`. Previously, Kong rebuilt the Lua environment, including a module reload, on every compile of a sandboxed function.

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ssary hash computation. Enabling the plugin now adds much less latency and CPU overhead to requests.

3

R

F

C

ce rename OLD_NAME NEW_NAME renames a workspace (cascading the new name to RBAC role-endpoint
oint-conflicts lists workspaces whose names collide with admin API URL prefixes.

F

P

- datakit:
  - Fixed an issue where the `response` node set `Content-Type: application/octet-stream` for `json_to_xml` output. It now defaults to `application/xml` when no explicit `Content-Type` is provided.
- grpc-gateway:
  - Fixed an issue where the plugin mapped the gRPC `UNIMPLEMENTED` status to HTTP 500 instead of 501. The new option `use_standard_grpc_status_codes` (default `off`) enables the fix.
- jwt-signer:
  - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.
- mocking:
  - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oauth2-introspection:
  - Fixed an authentication bypass where a revoked or expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false` result is now also cached briefly (5 seconds).
- openid-connect:
  - Fixed an issue where a request that authenticated but failed an authorization check (such as `scopes_required`) was not attributed to its consumer, consumer groups, or Kong Identity principal in logs and analytics.
  - Fixed an issue where the `redirect_uri` parameter was sent to the token endpoint for all grant types. It is now included only for the `authorization_code` grant, as required by RFC 6749.
  - The `ignore_signature` option now logs a warning when it is set, because it turns off JWS signature verification for the listed grants.
- opentelemetry:
  - Fixed an issue where the `kong.dp.cluster_cert.expiry` metric reused the same datapoint timestamp across exports, causing monitoring systems to treat the series as stale.
  - Reduced per-request overhead of `resource_attributes` by skipping template rendering entirely for static (non-templated) fields.
- prometheus:
  - A histogram's series (its buckets, `_count` and `_sum`) is now read as one atomic group during a scrape. Previously the exporter yielded and flushed output between individual samples, so an observation landing mid-scrape could bump `_count`/`_sum` after the buckets were already read, producing an internally inconsistent histogram in a single response.
  - Fixed an issue where `kong_nginx_requests_total` and `kong_nginx_connections_total` duplicated the `http` values under the `stream` subsystem. They are now exported once under `http`.
  - Histograms now expose every bucket (including those with a value of `0`) starting from the first observation of a given label set. Previously only buckets greater than or equal to the observed value were created, so lower buckets appeared lazily on later scrapes.
- saml:
  - Hardened signature validation against XML Signature Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that names the `ID` of the element it digests and consumes, so a signature can no longer be bound to a different element than the one that provides the identity. A response whose reference URI is empty or not a same-document fragment is now rejected.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a permanent failure that started when a broker answered an `ApiVersions` request with an error code, an empty API key set, or a malformed frame. The client cached that result and every later request failed with `ProduceRequest API version not negotiated; broker may be unreachable` until the node restarted, even after the broker recovered. The client now refuses an unusable nd retries the negotiation for a topic whose metadata is already cached.

onfluent-consume: Fixed the confluent schema registry client caching fetched schemas and subjects uthentication. Instances configured with different credentials could be served each other's cached lookups, uthentication once any authorized instance had warmed the cache. Cache entries are now scoped to the onfluent.ttl is now applied to these cache entries.

onfluent-consume: Fixed a crash that occurred when a plugin set oauth2 but omitted the oauth2_client record. Previously this returned a 500 from the Admin API during e fetching the schema registry OAuth2 token. The absent oauth2_client is now handled and falls back to out oauth2_client works end-to-end.

onfluent-consume: Fixed an issue where OAuth2 authentication against Confluent Cloud Schema d target-sr-cluster header.

method and query were reported incorrectly, or could raise an error, when a request hit the error handler

for 400, 404, 429, 502, and 504 were incorrect. The new configuration option use_standard_grpc_status_codes (default off) enables the fix.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Fixed an issue where a failed balancer creation leaked the per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.
- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.
- Fixed an issue where an in-flight request that completed after its upstream target became unavailable could restore the target to the latency balancer and route subsequent requests to it.
- Fixed an issue where custom Lua running with `untrusted_lua` set to `sandbox` could no longer create global variables or modify allowed library tables.
- Fixed an issue where custom streamed plugin schemas were still sandboxed when `untrusted_lua` was set to `on`, so schema code that needed to opt out of sandboxing (unlike custom plugin handlers, which already respected `on`) was incorrectly rejected on both the control plane and the data plane.
- Fixed an issue where `custom_validator` in schema definitions forwarded the enclosing schema as a second argument. This changed the call signature and could affect custom validators that declare more than one parameter. The single-argument call signature is restored.
- Fixed an issue where `/status/ready` could report a data plane ready before the new router and plugins iterator became active. Kong now reports readiness only after plugin configuration completes and the live objects are installed.
- Fixed an issue where the priority field of a cloned plugin could not be updated while it was still referenced by plugin instances.
- Fixed an issue where upstream failover could return intermittent 503 responses while a failover target remained healthy.
- Hardened HashiCorp Vault cloud authentication (AWS, Azure, GCP): Kong keeps credentials out of internal cache keys, truncates Vault error responses in logs, sets timeouts on all authentication requests, and applies `ssl_verify` to every outbound TLS call.
- **Konnect Only**. **Debugger**: Fixed an issue where deep copy of the worker logs stack overflowed by replacing `deep_copy` with `cycle_aware_deep_copy`.
- **PostgreSQL OAuth auth**: Refined the token cache key to be derived only from the fields that identify an OAuth configuration.

Clustering

- Fixed an issue where heavy packet loss could prevent the full sync from resuming.
- Fixed an issue where incremental config sync failed when an entity referenced a non-exported entity, for example rbac_users.
- Fixed an issue where incremental sync delta coalescing merged non-adjacent deltas of the same entity, which could drop an entity's earlier create and keep only a later update, reordering it after a delta that already referenced it (e.g. a plugin instance referencing a cloned plugin by name) and causing validation to fail with "plugin '' not enabled". Coalescing now only merges successive deltas of the same entity, preserving the causal order the data plane relies on.
- Fixed an issue where self-managed control planes rejected updates to a streamed plugin's handler while the plugin was referenced by a plugin instance.
- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.
- Fixed an issue where updates to a streamed plugin's definition (e.g. `handler`), or to a cloned plugin's definition (e.g. `priority`), were not applied on data planes when no referencing plugin instances changed. The data plane now rebuilds its plugins iterator so the updated definition takes effect, instead of continuing to execute the previous version.
- Fixed an issue where, with plugin streaming enabled, a single incremental sync validation failure would wipe out all plugin subschemas because the currently loaded plugins were not correctly excluded from the subschema unload.

Kong Manager

- Fixed an issue where the Export button on the Vitals Reports page did not download the report CSV.

D

C

es SCRAM channel binding failing to connect to a PostgreSQL server whose certificate is signed with

P

C

guration sync from the control plane, reducing the chance of a worker restart while the update is being

r version compatibility. This reduces CPU usage and transient memory pressure.

A

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Improved the performance of listing workspaces with `?counter=true`. Entity counts for the whole page are now retrieved with one grouped query per entity type instead of a separate count query per entity type per workspace, while keeping the counts exact.

Plugin

- opentelemetry:
  - Reduced memory usage of the access log queue by deferring OTLP payload conversion to export time.

Core

- Reduced config update latency on hybrid data planes by notifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in parallel.

3.15.0.4

Release date 2026/08/20

Feature

Plugin

- ai-proxy-advanced:
  - Added `cache_read_cost` and `cache_write_cost` to `model.options` so cached and cache-write prompt tokens can be priced per 1M tokens, alongside `input_cost` / `output_cost`.
  - Added filtering for empty Anthropic thinking blocks in request history.
  - Added support for reporting prompt cache token metrics in OpenAI-compatible format.
  - Added support for reporting prompt cache token metrics when using native (non-OpenAI) request/response formats.
  - Usage cost now supports per-cache-TTL cache-write pricing, context-window pricing factors, and service-tier pricing factors via `model.options` (`cache_write_cost_list`, `context_window_factor`, `service_tier_factor`). Anthropic (cache TTL + service tier), OpenAI, and Gemini service tiers are collected from upstream responses.
- ai-aws-guardrails, **ai-azure-content-safety**: Added `last_message` as a new optional value for `config.text_source`.

Bugfix

Plugin

- **ai guardrail plugins**:
  - Fixed issues where the ai-sanitizer plugin and ai-aws-guardrail plugin could not work properly with Claude code.
- ai-aws-guardrails:
  - Fixed an issue where the plugin ignored the configured `text_source` option when `allow_masking` was enabled, masking the entire raw request body instead.
- ai-prompt-guard:
  - Fixed an issue where the plugin failed to inspect requests sent in a provider's native format instead of the OpenAI-compatible format.
- ai-proxy-advanced:

d fail to decode when the client's request advertised Brotli support, since Anthropic's edge may then return
pression – which only understands gzip – can't inflate. Kong now normalizes the outgoing `Accept-`
LM provider: `gzip, identity` when the client accepts gzip, and `identity` otherwise, so a client that
ody on the paths that pass the upstream response through untouched.

uests using the compaction beta was undercounted. Kong now sums `usage.iterations` when present
kens / `output_tokens`, which only reflect non-compaction iterations.

e (billed when server-side tools such as grounding, code execution, or URL context are used) was not

ocks with omitted text were removed from request history.

ens and `cache_read_input_tokens` were excluded from prompt token counts, causing undercounting

ounted provisional `message_start` output tokens and summed cumulative `message_delta` usage.

25-06-27 Anthropic beta flag was dropped when routing to a Bedrock target, causing Bedrock to reject
error on failover from an Anthropic target.

ted as zero for Bedrock native format request to AWS models like Nova.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where Anthropic-format requests could report zero usage when it targets Bedrock's own model via conversion.
  - Fixed an issue where OpenAI, Azure, and Gemini realtime upstream URLs using `wss` were downgraded to `ws`.
  - Fixed an issue where a credential header could appear unredacted in access logs when using AWS temporary credentials with Bedrock.
  - Fixed an issue where Ollama rejected OpenAI message content arrays.
  - Fixed an issue where some Claude Code MCP tools did not correctly handle empty 'requires' array.
  - Fixed Gemini image generation token accounting to include reasoning tokens (`thoughtsTokenCount`) in `output_tokens` and report them as `output_tokens_details.reasoning_tokens`. Previously the reasoning tokens were dropped, which under-reported output tokens, analytics, and calculated cost for thinking-enabled image models.
  - Fixed Gemini realtime (Live API) token accounting to include reasoning tokens (`thoughtsTokenCount`) in the reported output tokens, and to report them as `output_tokens_reasoning_tokens_count`. Previously the reasoning tokens were dropped, which under-reported the output and total token counts of thinking-enabled realtime sessions.
  - Fixed token usage statistics being silently dropped for `audio/v1/audio/transcriptions` requests. Token-billed OpenAI/Azure transcription models (e.g. `gpt-4o-transcribe`) now report `input_tokens`/`output_tokens`/`total_tokens` and the `text_tokens`/`audio_tokens` breakdown in the analytics log when `logging.log_statistics` is enabled.

- ai-proxy:
  - Fixed Gemini token accounting to include reasoning tokens in `completion_tokens` for both OpenAI-format and native-format proxying.
- ai-sanitizer:
  - Fixed an issue where the input field for the llm/v1/responses route was not recognized.
- opentelemetry:
  - Fixed `gen_ai.client.token.usage` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
- prometheus:
  - Fixed `kong_ai_llm_tokens_total` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins' request extractors did not correctly honor the configured `text_source` option.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins failed to extract prompt text from OpenAI-native requests using the `input_text` content type.
- ai-proxy, **ai-proxy-advanced**: Fixed an issue where Gemini streaming analytics could over-report token usage. Gemini repeats its usage metadata on every stream chunk as a running request-level total, but the counts were summed across chunks instead of replaced, multiplying the reported usage by the number of chunks. Reasoning tokens were affected the most, because Gemini reports them in full on the first chunk and does not change them afterwards. A chunk that carried no usage metadata also reset the recorded output token count to zero.

CLI Command

- Hardened stale nginx PID detection to survive PID reuse. On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading `/proc/<pid>/cmdline`, so a reused PID from an ungraceful exit no longer blocks a restart.

C

nd start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with ssing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

3

R

E

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Security: Fixed CVE-2026-14916 by binding the JWT algorithm to the verification key's type in the `jwt_verify` node. This means that a symmetric algorithm can no longer be verified against an asymmetric key, preventing an authentication bypass where an attacker forges tokens by using the public key as an HMAC shared secret, under configurations where the node uses a PEM key string and `allowed_algorithms` is unset (any algorithm accepted).

- saml:
  - Security: Fixed CVE-2026-14917 in the SAML plugin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly, and logging a warning if misconfigured. This prevents a crafted unsigned SAML response from silently bypassing authentication when `validate_assertion_signature` is set to false.

Core

- Applied upstream nginx security patches for CVE-2026-42533.
- Applied upstream nginx security patches for CVE-2026-56434.
- Applied upstream nginx security patches for CVE-2026-60005.
- Applied upstream nginx security patches for CVE-2026-42055 and CVE-2026-48142.

Dependency

Core

- Bumped jq from 1.8.1 to 1.8.2.

3.15.0.2

Release date 2026/07/21

Bugfix

Plugin

- **Konnect Only**. **datakit**:
  - Added node source and runtime phase metadata to Datakit tracing events so Konnect Debugger can reconstruct the Datakit node graph from captured trace data.
- **otel**:
  - Fixed an issue where OpenTelemetry metrics only supported integer values by adding support for floating-point values in the OTLP exporter.
- ai-proxy:
  - Fixed OpenAI-compatible completions requests with string array prompts.
- ip-restriction:
  - Fixed an issue where warn log messages did not include the plugin name, making it difficult to identify the source of blocked requests.
- prometheus:
  - Histogram series are now emitted grouped by label set (all buckets in ascending `le` order, then `_count`, then `_sum`) so the output follows the OpenMetrics interleaving rule. Previously samples were grouped by suffix, which caused strict consumers such as Google Managed Prometheus to drop points.

Clustering

empty stat object instead of null before the first sync has run, keeping the response shape consistent.

P

serialization was not copied, causing data mutations when logs were generated concurrently via async

C

) leaked into the downstream phases (`header_filter`, `body_filter`, `response`, `log`). Dynamic ordering phases execute in static plugin-priority order.

events.

ecognized when configured in the `ordering` field of `plugins` on Data Planes.

r under sustained HTTPS traffic.

onse body of the external HTTP call `httpc:request()` was not captured. This includes the following changes: - Added a new span `kong.io.http.read_resp_body` to instrument to the `ready_body()` and `body_reader()` functions to capture the response body. -

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Added a `kong.3p.external_request_id` span attribute to the new `kong.io.http.read_resp_body` span and the `kong.io.http.request` span. - Added span linkage from the `kong.io.http.read_resp_body` span to the associated `kong.io.http.request` span. - Added an `external_request_id` attribute to the captured request/response headers/body items. - Skipped body capture upon low-level reading errors (e.g. network errors, timeouts, etc.) to avoid capturing corrupted data. - Improved external I/O latency measurement by removing the creation time of spans. - Moved capturing of request headers, body, and response headers from `kong.io.http.request_uri` span to the `kong.io.http.request` span.

- **Konnect Only**. **Debugger**: Fixed an issue where instrumenting third-party HTTP client calls produced two indistinguishable `kong.io.http.request` spans for `httpc:request_uri()`. It is now traced under a dedicated `kong.io.http.request_uri` span, and the `connect`, DNS, and internal `request` spans it triggers are nested beneath it. The `kong.io.http.request` span now also reports the `http.url` attribute.

Dependency

Core

- Bumped kong-redis-cluster from 1.6.0 to 1.6.1. This accepts the `x.y` Redis version format reported by some managed services (e.g. AWS ElastiCache Serverless Valkey) by making the patch component optional, and only warns on Redis version `9.0` or newer instead of `8.0` or newer.

Performance

Core

- Improved router update performance during incremental sync by updating service data in place instead of triggering a full router rebuild when only service attributes change.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

3.15.0.1

Release date 2026/07/07

Bugfix

Plugin

- acl:
  - Fixed an issue where the ACL plugin returned a 403 error for authenticated credentials with no consumer and no groups (e.g. ACE / Dev Portal application credentials) when configured with a deny list. A group-less identity cannot match a deny list and should be allowed through.
  - Fixed an issue where the ACL plugin ignored a consumer's ACL groups when `include_consumer_groups` was enabled and the consumer belonged to an enterprise consumer group, causing incorrect 403 denials (with `allow`) or incorrect access (with `deny`) for authenticated and anonymous consumers. The plugin now falls back to the consumer's ACL groups when no third-party authenticated groups are present.
- openid-connect:
  - Fixed an issue where a token endpoint response with `expires_in <= 0` was cached indefinitely with the proactive expiry re-check disabled, causing the plugin to silently forward an already-expired token to the upstream until a worker/process restart.
  - Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.
  - Fixed an issue where consumer groups could not be resolved when the groups claim was only available from the user info endpoint, which resulted in a `500` error.
- request-transformer:
  - Fixed a regression where `remove.headers` entries in `header:value` format (e.g. `correlationId:1222`) failed to remove the header. The value part is now correctly ignored during removal, restoring the behavior from 3.11.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

array was cleared before the asynchronous timer thread could safely encode it.

ng insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.

errors could be reported as subsequent socket write failures.

d after sandboxed template rendering in plugins such as `request-transformer` and `response-`

LS certificate setup failures instead of loading their backup configuration.

kit plugin whose `inputs` node contained a null value to an older data plane crashed the control plane's `index local 'value' (a userdata value)`.

Admin API

- Fixed an issue where resetting the password of an admin or developer failed with the error `Old password is invalid`, which happened when the credential was created before the basic auth hashing algorithm was upgraded from SHA1 to SHA256 (e.g. a `kong_admin` user created in an older version and carried over through an upgrade). The old password is now verified with the backward-compatible verification logic that supports both SHA1 and SHA256 hashed passwords.

Configuration

- Increased the default `proxy_ssl_verify_depth` and `lua_ssl_verify_depth` from `1` to `5`.

3.15.0.0

Release date 2026/07/02

Breaking Change

Core

- Without a configured license, the Admin API is now read-only, the same way it already behaves once a license has expired past its grace period. The proxy data plane keeps serving previously configured traffic. A small set of endpoints stays writable so the gateway can recover: `/licenses` to upload a new license, and `/keyring/recover`, `/keyring/import`, `/keyring/import/raw` so a keyring-encrypted license can be decrypted across restarts before the licensing subsystem comes up. Uploading a valid license restores full read-write access to the Admin API.
- Workspace names that shadow existing top-level admin API routes are no longer allowed. If you have an existing workspace name that conflicts with a admin route, it may be impacted. Check `GET /endpoints` to review the reserved first-level segments.

Feature

Core

- Added error attribution for plugin execution errors (HTTP 500). Kong classifies errors as `kong` (bundled plugins) or `custom` (custom plugins), and Prometheus exposes this via the `source` label to separate custom plugin failures from internal Kong errors.
- **Conditional Plugin Execution**: The plugin `condition` field now uses Common Expression Language (CEL) with broader operator and type support, aligned across UI, Control Plane, and Data Plane. Refer to the official documentation for the full syntax, supported variables, operators, and types.
- Added a Distroless Docker image variant for Kong API Gateway. The Distroless image ships only the Kong binary and its runtime dependencies, with no package managers, no shells or any other programs you would expect to find in a standard Linux distribution.
- Added a file system vault (`fs`).
- Added plugin cloning support. You can now clone existing bundled plugins, give them a new name, and optionally specify a priority. This is useful when you want to have multiple instances of a plugin with different configurations running in the same request. Currently, the cloneable plugins are: `openid-connect`, `acl`, `pre-function`, `post-function`, `request-transformer-advanced`, `request-transformer`, `response-transformer-advanced`, `response-transformer`, `key-auth`, `file-log`, `http-log`, `tcp-log`, `ip-restriction`, `route-by-header`, `opa`, `datakit`.
- Added plugin streaming support. You can now stream custom plugins dynamically from the control plane to its data planes. Refer to the documentation for more details on how to use this feature.
- Added support for `kong.router.get_route` and `kong.router.get_service` in the strict sandbox environment.
- Added the ability to reference the `cert` field of CA certificates.
- **analytics**: Added workspace information to the analytics payload.
- **vault**: Added a new `azure-certs` vault type for storing TLS certificates in Azure Key Vault.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

licit control over the Kafka consumer group ID.

-consume:

ge configuration option to surface Kafka client error messages in HTTP error responses. When enabled, is returned to the client instead of a generic message, aiding development and debugging. Defaults to

g field. When enabled, Kong returns the same HTTP status code the Lambda API returned (4xx or 5xx),

tity service with caching, consumer compatibility, and basic analytics.

- datakit:
  - Added a `non_nil` config attribute to the `property` node. When set to `true`, GET operations fail with an error if the property is unset, and SET operations fail with an error if the input is nil/null.
  - Added Konnect Debugger support for Datakit execution, including per-node spans, sanitized trace content capture with vault-aware redaction, and local debug trace responses using the same tracing event model with inline values.
- forward-proxy:
  - Added a `ca_certificates` option for per-plugin TLS verification. This setting allows certificate chain and hostname checks against specified CAs instead of using global settings, ensuring explicit trust.
- http-log:
  - Added opt-in outbound mTLS support via a new `client_certificate` config field.
- key-auth:
  - Added support for authentication via the Kong Identity service with caching, consumer compatibility, and basic analytics.
- oas-validation:
  - Added opt-in structured validation errors via the new `structured_errors` field, which provides structured error details. Use the `max_structured_errors` field (must be greater than 0) to cap the number of returned errors in structured format; when unset, all errors are returned. When `structured_errors` is disabled, the plugin preserves the original non-structured error format.
- openid-connect:
  - Added `proof_of_possession_mtls_from_header` for mTLS PoP validation via HTTP header when TLS is pre-terminated by downstream.
  - Added support for Kong Identity principal lookup after token verification, enabling consumer and consumer group mapping from Kong Identity with caching and miss-behavior control.
  - Added support for reading bearer tokens from a custom header.
  - Added the `cluster_cache_items` configuration option. This option controls which items are stored in the cluster cache backend configured via `cluster_cache_strategy`. Token material is AES-encrypted before being written to the cache.
  - Added `verify_signature` and `jwks_uri` fields to `token_exchange.subject_token_issuers` for verifying incoming subject token signatures before exchange.
- opentelemetry:
  - Added support for the `OTEL_RESOURCE_ATTRIBUTES` environment variable. You can now set custom resource attributes for OpenTelemetry instrumentation as secondary resource attributes to the `resource_attributes` configuration.
- rate-limiting-advanced:
  - Added the `counter_key` field to let you choose which consumer attribute is now used as the rate-limiting counter identifier. Supported values are `consumer.id`, `consumer.username`, and `consumer.custom_id`.
- ace Added Kong Identity principal fetching after authentication, enabling consumer and consumer group mapping from the Konnect identity service.
- kafka-upstream, **confluent**: Added support for forwarding HTTP request headers as native Kafka record headers.
- kafka-upstream, **kafka-log**, **kafka-consume**, **confluent**, **confluent-consume**: Added SASL OAUTHBEARER authentication support.
- solace-upstream, **solace-log**, **solace-consume**: Added support for OAuth2 client credentials authentication. Configured with the `CLIENT_CREDENTIALS` scheme, tokens are cached and renewed automatically.

Clustering

payloads larger than the configured `cluster_max_payload` size to each other, which can help avoid
nprove performance. When this happens, Kong now emits a log entry containing the config payload size

P

uest headers parsed from raw request headers for HTTP/1.x requests.

K

tom plugins, to the new FreeForm renderer.

F

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where we never stored the response model in the streaming path. It always fell back to the request model.

- **ai-guardrail**:
  - Fixed an issue where latency metrics were missing in the Prometheus output.
- **ai-guardrails**:
  - Fixed an issue where multiple guardrail plugins couldn't function together.
- **ai-transformers**:
  - Fixed an issue where `ai-request-transformer` and `ai-response-tranformer` plugins did not work properly with the `ollama` provider.
- **kafka-upstream, kafka-log, kafka-consume, confluent, confluent-consume**:
  - Fixed Kafka keepalive pools leaking connections across different SASL credentials and TLS verification settings.
- acl:
  - Fixed an issue where anonymous consumers with valid legacy ACL group credentials were rejected with a 403 error when `include_consumer_groups` was enabled and no Enterprise consumer groups were assigned.
- ai-a2a-proxy:
  - Reassembled SSE `data:` events that span multiple `body_filter` chunks and dropped oversized events safely, preventing `sse_last_event` type errors that crashed the request on large agent responses.
- ai-azure-content-safety:
  - Fixed a crash (HTTP 500) that occurred when `azure_use_managed_identity` was enabled.
  - Fixed a bug where the plugin returned HTTP 500 when the LLM response content was empty. Empty content now skips the content-safety check and passes through.
  - Fixed an issue where the response guard was sending the entire JSON response body to Azure Content Safety instead of just the assistant message content, which caused false positives when response metadata fields matched blocklist entries.
- ai-mcp-oauth2:
  - Fixed an issue where token introspection crashed when `client_auth` was set to `private_key_jwt` and `client_jwk` was provided as a JSON string.
  - Fixed an issue where downstream scope-based ACL evaluation used claims from the inbound token instead of the exchanged token when token exchange was enabled.
  - Fixed an issue where token exchange actor tokens were not sourced correctly from `token_exchange.request`, ensuring correct forwarding of actor tokens configured in headers or plugin config.
  - Fixed an issue where `token_exchange.cache.enabled = false` was ignored and exchanged tokens were still cached because the cache toggle incorrectly read `token_exchange.cache.ttl` instead of `token_exchange.cache.enabled`.
- ai-mcp-proxy:
  - Fixed an issue where raw non-200 upstream responses triggered misleading MCP body parse warnings.
  - Fixed an issue where the `WWW-Authenticate: Bearer error="insufficient_scope"` header was not emitted.
  - Fixed an issue where internal unix-socket subrequests lost the original client IP address, causing IP-aware plugins like IP Restriction to reject legitimate requests.
  - Fixed an issue where, since 3.14, call tools converted from API could fail when Kong used a self-signed certificate.
  - Fixed an issue where the tool ID might be repeated.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

erver responses were rewritten to empty objects when the plugin re-encoded the body for ACL list filtering.

e.g. `deck sync`), causing `tools/list` to return empty results.

rmat requests (gemini, anthropic, bedrock, cohere, huggingface) by forwarding an OpenAI-shaped body

A version of Realtime API.

g requests were not mapped consistently across Anthropic, Gemini, Bedrock, and other providers.

Code were truncated when doing non-passthrough proxy.

ormat for the Databricks provider.

e's token usage if available.

  - Fixed an issue where we didn't collect the model name in OpenAI streaming response from Azure.

- Fixed an issue where we didn't correctly ungzip streaming responses.
- Fixed an issue where dashscope replied with a 500 error to inference requests.
- Fixed an issue where GenAI spans were not finished sometimes.
- Fixed an issue where observability errors occurred when `gpt-image-1.5` returned image token usage details.
- Fixed an issue where Azure OpenAI authentication failed during load balancer failover due to token refresh attempts in `balancer_by_lua`. Tokens are now pre-fetched in the `access` phase.
- Fixed an issue where concurrent worker processes could race to initialize the semantic routing vectorDB index, causing `"Index already exists"` errors on startup or config reload.
- Fixed an issue where we treated plaintext Anthropic response as gzipped.
- Fixed an issue where `output_config` was rejected by Vertex AI in Anthropic-native requests.
- Fixed an issue where the configured model name may never be used for the Azure provider.
- Fixed an issue where Claude Code (or CLI) statistics were missing when using an Anthropic model via the Bedrock provider.

- ai-proxy:
  - Fixed an issue with the Gemini driver dropping Gemini-native fields (e.g. `cachedContent`, `safetySettings`) and Vertex AI-only fields (e.g. `labels`) during OpenAI-to-Gemini request transformation.
  - Fixed an issue where `options.upstream_url` was not respected in native mode.
- ai-rate-limiting-advanced:
  - Emitted the documented `X-AI-RateLimit-*` headers on `429` responses from policies that partition by `provider` or `model`.
  - Fixed an issue with token counters being decremented on AI Semantic Cache hits.
  - Fixed an issue where `model` or `provider` matchers without `partition_by` were skipped, causing rate limits to be missed.
  - Fixed policy integration with `ai-proxy` for single-target and multi-target AI proxy routes.
- ai-sanitizer:
  - Fixed an issue where the AI Sanitizer plugin could not parse the request body correctly.
- ai-semantic-response-guard:
  - Fixed an issue where the plugin failed to populate Redis with rule documents due to using worker ID gating instead of worker mutex for vectordb operations.
- aws-lambda:
  - Fixed an issue where empty JSON array bodies (`[]`) were incorrectly converted to objects (`{}`) when using `is_proxy_integration=true` and `empty_arrays_mode=correct` alongside the Exit Transformer plugin.
  - Fixed an issue where a 500 error occurred when `is_proxy_integration` was enabled and the Lambda function returned `null` values in `headers` or `multiValueHeaders`. Null values are now converted to empty strings.
- datakit:
  - Added Lua control plane compatibility checks to prevent unsupported Datakit configurations from being pushed to older data planes in hybrid mode.
- exit-transformer:
  - Fixed an issue where a response body that was a JSON primitive (e.g. `false`) caused a 500 internal error.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

dvanced plugin's API threw 500 errors when handling HTTP PUT requests.

om unix socket incorrectly.

tokens without dots) caused a 500 Internal Server Error instead of returning 401 Unauthorized.

memory leaks.

blocked on identity_realms.

  - Fixed an issue where `$ref` dereferencing failed with `invalid ref: <code>` when bare-numeric keys were used under `components.responses`. Now supports numeric-key fallback.

- oas-validation:
  - Fixed an issue where the OAS specification cache failed to distinguish between different values of `api_spec_encoded` for the same specification content.
  - Fixed an issue where path parameters were extracted from the wrong URI segment when `custom_base_path` was used with bare `/{param}` paths.
- openid-connect:
  - Fixed an inconsistency where `credential_claim` resolved the same claim name to a different value than `upstream_headers_claims` because it consulted the `id_token` before the `access_token`, while the upstream headers resolver consulted the `access_token` first. With IdPs (e.g. Okta) that set distinct `sub` values in the two tokens, `authenticated_entity.id` did not match the upstream header the plugin set on the same request.
  - Fixed an issue where nested claim paths were not resolved correctly.
  - Fixed an issue where Redis-backed session storage could fail when Redis connections were routed through a proxy.
  - Fixed an issue where token cluster cache entries were not written when the `cluster_cache_strategy` was updated using incremental sync.
  - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.
- opentelemetry:
  - Fixed an issue where request-scoped `resource_attributes` could leak across concurrent queued `traces`, `logs`, and access log exports when the plugin was configured globally.
  - Fixed an issue where template `resource_attributes` in plugin configuration didn't work as expected for metrics.
  - Fixed an issue where the plugin crashed when DNS or balancer failures occurred during upstream retries.
- proxy-cache-advanced:
  - Fixed an issue where the cache management Admin API did not work when the `redis` strategy configuration was provided via a partial.
- rate-limiting:
  - Fixed an issue where the sync timer did not use updated configuration.
- request-transformer:
  - Fixed a regression where `remove.body` entries in `key:value` format (e.g. `scope:USER`) failed to remove the field from the request body. The value part is now correctly ignored during removal, restoring the behavior from 3.11.
- request-validator:
  - Added `array_length_compat` config option to allow legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.
- service-protection:
  - Fixed an issue where the Service Protection plugin could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after Rate Limiting Advanced and only evaluates requests that pass rate limiting, by updating its priority from 915 to 901.
- solace-upstream:
  - Fixed an issue where requests under high concurrency could fail with connection timeouts despite a sufficient `connect_timeout` being configured.
- ai-semantic-prompt-guard, **ai-semantic-response-guard**: Fixed an issue where full-sync reconfigure rebuilt semantic vectorDB indexes even when the plugin configuration was unchanged.
- Fixed AI Proxy Advanced Gemini failover handling when a target sets `upstream_url` to `ngx.null`, avoiding an empty trailing query string in the computed upstream URL.

epalive pool, added SSL handshake retry, and improved error propagation.

nts.

0701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.

maximum number of headers (CVE-2026-49975).

the status code of connections that were closed by the client.

nts were incorrectly included in Redis spans.

apture consumed streaming response chunks.

oncatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **Debugger**: Fixed an issue where the time spent during the TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side latency.
- **Debugger**: Fixed an issue where the default key name `apikey` of the `key-auth` and `key-auth-enc` plugins was not redacted when captured by the Debugger. This has been resolved by adding `apikey` to the list of default redacted headers.
- Fixed a bug where tables in `strict` and `lax` `untrusted_lua` modes were not properly protected against modification.
- Fixed an issue where anonymous reports omitted model usage when one provider used multiple models.
- Fixed an issue where Kong Manager users authenticated through IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim changes, or end up with inconsistent mappings during concurrent authentication requests.
- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. It now includes a cert fingerprint for proper connection isolation.
- Fixed an issue where `$remote_addr` was not replaced with the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` is enabled on the proxy listener. Also added proper error body messages for 501 and 505 status codes.
- Fixed an issue where the `KONG_ANALYTICS_FLUSH_INTERVAL` environment variable was not correctly applied to the analytics flush interval.
- Fixed an issue where the real client IP from Proxy Protocol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 plain HTTP to HTTPS port) or when the client disconnected prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from `$proxy_protocol_addr` on early request termination, and added `error_page` 497 routing through `kong_error_handler`.
- Fixed an issue where the request context could inherit metatables from previous TLS connections, leading to cross-request context leakage and increased memory usage.
- Fixed CVE-2026-6338.
- **router**: Fixed an issue where WebSocket routes (`ws`/`wss`) returned HTTP 404 when using `router_flavor=expressions` due to incorrect protocol matching expressions being generated.
- **sandbox**: Fixed an issue where the `-1` error level was incorrectly passed to `error()`. The error level is now `0`, ensuring sandbox error messages are reported consistently regardless of traceback depth.
- **websocket**: Fixed an issue where the response body from an upstream was silently dropped when the upstream rejected the WebSocket handshake with a non-101 HTTP status code (e.g. 400, 401, 403). The body is now read from the TCP socket before it is closed and forwarded to the client, supporting both `Content-Length` and `Transfer-Encoding: chunked` framing.

Clustering

- Fixed an issue where a config backup exported by a data plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or `oic_issuers`. The export incorrectly included a `ws_id` field for those entities, which the declarative config schema rejects as an unknown field.
- Fixed an issue where a Data Plane failed to load config updates when a CA certificate expired. The Data Plane now accepts the config and logs a warning.
- Fixed an issue where DP syncing failed with large configs. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful full-sync.
- Fixed an issue where hybrid data planes retried unrecoverable CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable reconnect storms.
- Fixed an issue where incremental DP (sync v2) would not export config to fallback storage when the control plane used periodic polling instead of `notify_new_version`.
- Fixed an issue where incremental sync (ICS) would crash and stop further sync attempts.
- Fixed an issue where paginated full sync could keep stale pending foreign-key references across a failed resume, causing spurious validation failures.
- Fixed an issue where the DP may spawn multiple sync timers when downgrading sync protocol to v1, which may cause excessive load on CP and lead to

lling pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

P

ery_arg()` would destructively rewrite unrelated query arguments by normalizing `+` to `%20` and stripping
arguments are now preserved verbatim.

stomized by array indices, like `kong.upstream.try[1].balancer_latency_ns: 'return 140545'` or
ve integers are supported as array indices.

ction incorrectly parsed the `Content-Disposition` name parameter if it wasn't double-quoted.

C

t respect the command `prefix` argument.

D

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **jwt-signer**: Fixed an issue where the plugin failed to load configuration and errored when `null` or `nil` values were passed to `access_token_keyset` or `channel_token_keyset`.

Admin API

- Kong no longer allows using a license referenced by a vault to be applied via the Admin API.
- **RBAC**: Fixed an issue where `PATCH /rbac/roles/{name_or_id}/entities/{entity_id}` returned a misleading `404` response instead of the correct error (e.g. a `400` schema violation) when the update failed.

Kong Manager

- Fixed an issue where default workspace roles failed to be created on workspace creation.
- Fixed workspace navigation for workspace names containing non-ASCII characters.

Dependency

Core

- Bumped avro-c from 1.21.0 to 1.21.1.
- Bumped brotli from 1.1.0 to 1.2.0.
- Bumped jansson from 2.14.1 to 2.15.0.
- Bumped kong-gql from 0.2.3 to 0.2.4 to fix the null value issue in the GraphQL query syntax that caused a parsed AST error in DeGraphQL plugin.
- Bumped kong-lua-resty-ljsonschema from 1.3.1 to 1.3.3.
- Bumped kong-pgmoon from 2.3.3.0 to 2.3.4.0, which adds SHA-384 support for SCRAM channel binding during PostgreSQL SSL authentication.
- Bumped kong-redis-cluster from 1.5.7 to 1.6.0.
- Bumped libexpat from 2.8.0 to 2.8.1.
- Bumped libxml2 from 2.15.1 to 2.15.3.
- Bumped libxslt from 1.1.43 to 1.1.45.
- Bumped lua-resty-gcp from 0.0.19 to 0.0.20 to add support for proxy options.
- Bumped lua-resty-healthcheck from 3.1.1 to 3.1.2
- Bumped lua-resty-mail from 1.1.0 to 1.2.0.
- Bumped lua-resty-openssl from 1.7.0 to 1.7.1.
- Bumped lua-resty-websocket from version 0.4.0.2 to 0.5.0 to solve an issue when handling protocol upgrade failure.
- Bumped LuaRocks from 3.12.2 to 3.13.0.
- Bumped multipart from 0.5.11 to 0.5.12.
- Bumped OpenResty from 1.27.1.2 to 1.29.2.5.
- Bumped OpenSSL from 3.4.3 to 3.5.7.
- Bumped penlight from 1.14.0 to 1.15.0.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

C

2026-03-19.

certs) from Kong packages.

P

C

oducing a shared payload pool and reducing the default local buffer size.

P

- Reduced the OTel trace-pipeline CPU overhead.

3.14.0.14

Release date 2026/08/26

Feature

Core

- Enabled TLS 1.3 for the FIPS cipher suite in the FIPS-140-3 artifacts, and configured those builds to use AES-based TLS 1.3 ciphersuites. The FIPS-140-2 artifacts keep TLS 1.2.
- Kong now publishes one artifact per FIPS certification level. The `kong-enterprise-edition-fips` package and the `<version>-<distro>-fips` image keep the OpenSSL FIPS provider 3.0.9 (FIPS 140-2). The new `kong-enterprise-edition-fips-140-3` package and the `<version>-<distro>-fips-140-3` image use the provider 3.1.2 (FIPS 140-3).
- The FIPS-140-3 artifacts serve an explicit TLS 1.2 cipher list: four ECDHE suites with AES-GCM. The FIPS-140-2 artifacts keep the OpenSSL FIPS alias, which also admits static RSA key transport and SSLv3-era suites. To serve a different list, set `ssl_cipher_suite = custom` and `ssl_ciphers`.

Plugin

- openid-connect:
  - KONG_FIPS=on no longer generates default EdDSA JWKS keys. Use an RSA-PSS or NIST-curve ECDSA key instead.

    If you migrate from the FIPS package to the FIPS-140-3 package, this change breaks existing EdDSA-based configurations.

Bugfix

Plugin

- datakit:
  - Fixed an issue where the `response` node set `Content-Type: application/octet-stream` for `json_to_xml` output. It now defaults to `application/xml` when no explicit `Content-Type` is provided.
- jwt-signer:
  - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.
- mocking:
  - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oauth2-introspection:
  - Fixed an authentication bypass where a revoked or expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false` result is now also cached briefly (5 seconds).
- openid-connect:
  - Fixed an issue where a request that authenticated but failed an authorization check (such as `scopes_required`) was not attributed to its consumer, or consumer groups in logs and analytics.
  - The `ignore_signature` option now logs a warning when it is set, because it turns off JWS signature verification for the listed grants.
- opentelemetry:

exported on data planes whose configuration is rebuilt more frequently than the metrics `push_interval`.
ed and recreated on every configuration rebuild, so its `push_interval` countdown is preserved.

`expiry` metric reused the same datapoint timestamp across exports, causing monitoring systems to treat the

`utes` by skipping template rendering entirely for static (non-templated) fields.

`l` and `kong_nginx_connections_total` duplicated the `http` values under the `stream` subsystem. They are

ure Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that
s, so a signature can no longer be bound to a different element than the one that provides the identity. A
me-document fragment is now rejected.

confluent-consume: Fixed a permanent failure that started when a broker answered an `ApiVersions`
lformed frame. The client cached that result and every later request failed with `ProduceRequest API`

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`version not negotiated; broker may be unreachable` until the node restarted, even after the broker recovered. The client now refuses an unusable `ApiVersions` response, keeps the previous negotiation, and retries the negotiation for a topic whose metadata is already cached.

- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed an issue where OAuth2 authentication against Confluent Cloud Schema Registry failed because the client did not send the required `target-sr-cluster` header.

PDK

- Fixed an issue on HTTP/2 connections where the request method and query were reported incorrectly, or could raise an error, when a request hit the error handler such as a 502 from an upstream connection reset.
- Fixed an issue where the array log entries could not be customized by array indices, like `kong.upstream.try[1].balancer_latency_ns: 'return 140545'` or `x.y.foo[2]: "return [[how are you?]]"`. Only positive integers are supported as array indices.

Core

- Fixed an issue where a failed balancer creation leaked the per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.
- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.
- Fixed an issue where an in-flight request that completed after its upstream target became unavailable could restore the target to the latency balancer and route subsequent requests to it.
- Fixed an issue where `/status/ready` could report a data plane ready before the new router and plugin iterator became active. Kong now reports readiness only after plugin configuration completes and the live objects are installed.
- Fixed an issue where upstream failover could return intermittent 503 responses while a failover target remained healthy.
- Hardened HashiCorp Vault cloud authentication (AWS, Azure, GCP): Kong keeps credentials out of internal cache keys, truncates Vault error responses in logs, sets timeouts on all authentication requests, and applies `ssl_verify` to every outbound TLS call.
- **Konnect Only**. **Debugger**: Fixed a stack overflow in the deep copy of worker logs. The fix replaces `deep_copy` with `cycle_aware_deep_copy`.

Clustering

- Fixed an issue where heavy packet loss could prevent the full sync from resuming.
- Fixed an issue where incremental config sync failed when an entity referenced a non-exported entity, for example rbac_users.
- Fixed an issue where incremental sync delta coalescing merged non-adjacent deltas of the same entity, which could drop an entity's earlier create and keep only a later update, reordering it after a delta that already referenced it (e.g. a plugin instance referencing a cloned plugin by name) and causing validation to fail with "plugin ‘' not enabled". Coalescing now only merges successive deltas of the same entity, preserving the causal order the data plane relies on.
- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.
- Fixed an issue where, with plugin streaming enabled, a single incremental sync validation failure would wipe out all plugin subschemas because the currently loaded plugins were not correctly excluded from the subschema unload.

Kong Manager

- Fixed an issue where the Export button on the Vitals Reports page did not download the report CSV.

Dependency

C

0-3 artifacts. The FIPS-140-2 artifacts keep the provider 3.0.9.

es SCRAM channel binding failing to connect to a PostgreSQL server whose certificate is signed with

P

C

guration sync from the control plane, reducing the chance of a worker restart while the update is being

r version compatibility. This reduces CPU usage and transient memory pressure.

A

unter=true. Entity counts for the whole page are now retrieved with one grouped query per entity type instead of a separate count query per entity type per workspace, while keeping the counts exact.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- opentelemetry:
  - Reduced memory usage of the access log queue by deferring OTLP payload conversion to export time.

Core

- Reduced config update latency on hybrid data planes by notifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in parallel.

3.14.0.13

Release date 2026/08/19

Feature

Core

- Added a Distroless Docker image variant for Kong API Gateway. The Distroless image ships only the Kong binary and its runtime dependencies, with no package managers, no shells, and no other programs you would expect to find in a standard Linux distribution.

Plugin

- ai-proxy-advanced:
  - Added `cache_read_cost` and `cache_write_cost` to `model.options` so cached and cache-write prompt tokens can be priced per 1M tokens, alongside `input_cost` / `output_cost`.
  - Added filtering for empty Anthropic thinking blocks in request history.
  - Added support for reporting prompt cache token metrics in OpenAI-compatible format.
  - Added support for reporting prompt cache token metrics when using native (non-OpenAI) request/response formats.
  - Usage cost now supports per-cache-TTL cache-write pricing, context-window pricing factors, and service-tier pricing factors via `model.options` (`cache_write_cost_list`, `context_window_factor`, `service_tier_factor`). Anthropic (cache TTL + service tier), OpenAI, and Gemini service tiers are collected from upstream responses.

Bugfix

Plugin

- ai-proxy-advanced:
  - Fixed a bug where a response from Anthropic could fail to decode when the client's request advertised Brotli support, since Anthropic's edge may then return a Brotli-encoded body that Kong's response decompression – which only understands gzip – can't inflate. Kong now normalizes the outgoing `Accept-Encoding` request header before proxying to any LLM provider: `gzip, identity` when the client accepts gzip, and `identity` otherwise, so a client that cannot inflate gzip is never handed a compressed body on the paths that pass the upstream response through untouched.
  - Fixed an issue where native Anthropic streaming counted provisional `message_start` output tokens and summed cumulative `message_delta` usage.
  - Fixed an issue where output token usage was reported as zero for Bedrock native format request to AWS models like Nova.
  - Fixed an issue where Anthropic-format requests could report zero usage when it targets Bedrock's own model via conversion.

ealtime upstream URLs using `wss` were downgraded to `ws`.

uests using the compaction beta was undercounted. Kong now sums `usage.iterations` when present
kens / `output_tokens`, which only reflect non-compaction iterations.

e (billed when server-side tools such as grounding, code execution, or URL context are used) was not

cks with omitted text were removed from request history.

s did not correctly handle empty 'requires' array.

o include reasoning tokens (`thoughtsTokenCount`) in `output_tokens` and report them as
viously the reasoning tokens were dropped, which under-reported output tokens, analytics, and calculated

g to include reasoning tokens (`thoughtsTokenCount`) in the reported output tokens, and to report them as
busly the reasoning tokens were dropped, which under-reported the output and total token counts of

for `audio/v1/audio/transcriptions` requests. Token-billed OpenAI/Azure transcription models (e.g.
tput_tokens/total_tokens` and the `text_tokens/audio_tokens` breakdown in the analytics log when

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`logging.log_statistics` is enabled.

- ai-proxy:
  - Fixed Gemini token accounting to include reasoning tokens in `completion_tokens` for both OpenAI-format and native-format proxying.
- opentelemetry:
  - Fixed `gen_ai.client.token.usage` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
- prometheus:
  - Fixed `kong_ai_llm_tokens_total` never reporting token counts for non-text-generation AI categories (image, video, realtime, and audio transcription generation), because it only read the legacy `prompt_tokens`/`prompt_tokens_details` analytics log fields instead of falling back to the `input_tokens`/`input_tokens_details` fields those categories are logged under.
- ai-proxy, **ai-proxy-advanced**: Fixed an issue where Gemini streaming analytics could over-report token usage. Gemini repeats its usage metadata on every stream chunk as a running request-level total, but the counts were summed across chunks instead of replaced, multiplying the reported usage by the number of chunks. Reasoning tokens were affected the most, because Gemini reports them in full on the first chunk and does not change them afterwards. A chunk that carried no usage metadata also reset the recorded output token count to zero.

CLI Command

- Hardened stale nginx PID detection to survive PID reuse. On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading `/proc/<pid>/cmdline`, so a reused PID from an ungraceful exit no longer blocks a restart.

Core

- On shell-free (distroless) images, the pluginserver query and start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with a clear error on an unterminated quote or `${`, instead of passing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

3.14.0.12

Release date 2026/08/10

Bugfix

Plugin

- ai-mcp-oauth2:
  - Fixed a JWT security vulnerability.
- datakit:
  - Fixed CVE-2026-14916 by binding the JWT algorithm to the verification key's type in the `jwt_verify` node, so a symmetric algorithm can no longer be verified against an asymmetric key. This prevents an authentication bypass where an attacker forges tokens by using the public key as an HMAC shared secret, under configurations where the node uses a PEM key string and `allowed_algorithms` is unset (any algorithm accepted).
- saml:
  - Security: Fixed CVE-2026-14917 in the SAML plugin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly (and logging a warning if misconfigured), preventing a crafted unsigned SAML response from silently bypassing authentication under configurations that set `validate_assertion_signature` to false.

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

2533.

6434.

0005.

2055 and CVE-2026-48142.

D

C

3

R

Feature

Plugin

- ai-aws-guardrails, **ai-azure-content-safety**: Added `last_message` as a new optional value for `config.text_source`.

Bugfix

Plugin

- ai-aws-guardrails:
  - Fixed an issue where the plugin ignored the configured `text_source` option when `allow_masking` was enabled, masking the entire raw request body instead.
- ai-prompt-guard:
  - Fixed an issue where the plugin failed to inspect requests sent in a provider's native format instead of the OpenAI-compatible format.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins' request extractors did not correctly honor the configured `text_source` option.
- ai-aws-guardrails, **ai-azure-content-safety**, **ai-custom-guardrail**, **ai-gcp-model-armor**, **ai-lakera-guard**, **ai-prompt-guard**, **ai-sanitizer**, **ai-semantic-prompt-guard**, **ai-semantic-response-guard**: Fixed an issue where AI Gateway guardrail plugins failed to extract prompt text from OpenAI-native requests using the `input_text` content type.
- ai-sanitizer, **ai-aws-guardrails**: Fixed an issue where the plugins were prevented from working properly with Claude Code.

Core

- Fixed an issue where custom Lua running with `untrusted_lua` set to `sandbox` could no longer create global variables or modify allowed library tables.

3.14.0.10

Release date 2026/07/29

Feature

Plugin

- ai-a2a-proxy:
  - now rewrites the URL for all `supportedInterfaces` in the agent card.

Bugfix

Plugin

- ai-azure-content-safety:
  - Fixed a bug where the plugin returned HTTP 500 when the LLM response content was empty. Empty content now skips the content-safety check and passes through.
- ai-proxy-advanced:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

vas not treated as a system prompt when transforming requests to Bedrock and Gemini providers.

sage content arrays.

s sent as a Bedrock/Gemini system message which only support string.

g. `api-key`, `x-api-key`) leaked the provider credential into access logs instead of being redacted.

25-06-27 Anthropic beta flag was dropped when routing to a Bedrock target, causing Bedrock to reject error on failover from an Anthropic target.

(e.g. `cachedContent`, `safetySettings`) and Vertex AI-only fields (e.g. `labels`) during OpenAI-to-

without `partition_by` were skipped, causing rate limits to be missed.

  - Fixed an issue where the input field for the llm/v1/responses route was not recognized.

- ip-restriction:
  - Fixed an issue where warn log messages did not include the plugin name, making it difficult to identify the source of blocked requests.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a crash that occurred when a plugin set `schema_registry.confluent.authentication.mode` to `oauth2` but omitted the `oauth2_client` record. Previously this returned a 500 from the Admin API during validation, and once saved it also crashed at runtime while fetching the schema registry OAuth2 token. The absent `oauth2_client` is now handled and falls back to the default authentication method, so a configuration without `oauth2_client` works end-to-end.

Clustering

- **Clustering**: Improved RPC connection stability during long-running calls: a data plane now cancels a timed-out RPC so a late response cannot disrupt later calls, and the connection is kept alive (within a bound) while calls are still in flight instead of being dropped as a missed heartbeat.
- **Clustering**: The `/status/sync` endpoint now returns an empty stat object instead of null before the first sync has run, keeping the response shape consistent.

PDK

- Fixed a race condition issue where the `tries` field of log serialization was not copied, causing data mutations when logs were generated concurrently via async timers.
- Fixed an issue where dynamic upstream TLS settings set via the PDK were lost during buffered proxying.

Core

- **Konnect Only**. **Debugger**: Fixed an error that could occur under sustained HTTPS traffic.
- **Konnect Only**. **Debugger**: Fixed an issue where the response body of the external HTTP call `httpc:request()` was not captured.
- **PostgreSQL OAuth auth**: Refined the token cache key to be derived only from the fields that identify an OAuth configuration.

Performance

Core

- Improved router update performance during incremental sync by updating service data in place instead of triggering a full router rebuild when only service attributes change.

3.14.0.9

Release date 2026/07/14

Bugfix

Plugin

- **otel**:
  - Fixed an issue where OpenTelemetry metrics only supported integer values by adding support for floating-point values in the OTLP exporter.
- acl:
  - Fixed an issue where the ACL plugin returned 403 for authenticated credentials with no consumer and no groups (e.g. ACE / Dev Portal application credentials) when configured with a deny list. A group-less identity cannot match a deny list and should be allowed through.
- ai-mcp-proxy:

ollisions.

eaders correctly.

g dropped string message content in Kong’s internal format, causing other plugins to receive invalid data.

ployment-style routing.

handle the OpenAI `content_parts` format in system messages, user/assistant messages, and array-typed

th string array prompts.

hantic Cache hits.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- openid-connect:

  - Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.
  - Fixed an issue where consumer groups could not be resolved when the groups claim was only available from the user info endpoint, which resulted in a `500` error.
- prometheus:
  - A histogram's series (its buckets, `_count` and `_sum`) is now read as one atomic group during a scrape. Previously, the exporter yielded and flushed output between individual samples, so an observation landing mid-scrape could bump `_count`/`_sum` after the buckets were already read, producing an internally inconsistent histogram in a single response.
  - Histogram series are now emitted grouped by label set (all buckets in ascending `le` order, then `_count`, then `_sum`) so the output follows the OpenMetrics interleaving rule. Previously, samples were grouped by suffix, which caused strict consumers such as Google Managed Prometheus to drop points.
  - Histograms now expose every bucket (including those with a value of `0`) starting from the first observation of a given label set. Previously, only buckets greater than or equal to the observed value were created, so lower buckets appeared lazily on later scrapes.
- request-transformer:
  - Fixed a regression where `remove.headers` entries in `header:value` format (e.g. `correlationId:1222`) failed to remove the header. The value part is now correctly ignored during removal, restoring the behavior from 3.11.

Core

- **Debugger**: Fixed an issue where instrumenting third-party HTTP client calls produced two indistinguishable `kong.io.http.request` spans for `httpc:request_uri()`. It is now traced under a dedicated `kong.io.http.request_uri` span, and the `connect`, DNS, and internal `request` spans it triggers are nested beneath it. The `kong.io.http.request` span now also reports the `http.url` attribute.
- Fixed an issue where dynamic plugin ordering (`ordering`) leaked into the downstream phases (`header_filter`, `body_filter`, `response`, `log`). Dynamic ordering now again applies only to the `access` phase; downstream phases execute in static plugin-priority order.

Admin API

- Fixed an issue where resetting the password of an admin or developer failed with the error `Old password is invalid`, which happened when the credential was created before the basic auth hashing algorithm was upgraded from SHA1 to SHA256 (e.g. a `kong_admin` user created in an older version and carried over through an upgrade). The old password is now verified with the backward-compatible verification logic that supports both SHA1 and SHA256 hashed passwords.

Dependency

Core

- Bumped kong-redis-cluster from 1.5.7 to 1.6.1. This fixes a breaking change in `refresh_slots` that could lead to a lock issue, optimizes hot paths (slot iteration and array creation), and avoids early concatenation of error log strings. This also accepts the `x.y` Redis version format reported by some managed services (e.g. AWS ElastiCache Serverless Valkey) by making the patch component optional, and only warns on Redis version `9.0` or newer instead of `8.0` or newer.

3.14.0.8

Release date 2026/07/02

Bugfix

Plugin

- acl:
  - Fixed an issue where the ACL plugin ignored a consumer's ACL groups when `include_consumer_groups` was enabled and the consumer belonged to an
    denials (with `allow`) or incorrect access (with `deny`) for authenticated and anonymous consumers. The
    s when no third-party authenticated groups are present.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

mitted the protected resource metadata URL when a token was provided.

dvertised unsupported SSE streams.

ntication probed `metadata.google.internal` when a service account JSON was configured.

edrock Anthropic format requests using ARN model IDs.

am requests using ARN model IDs could fail when handling streamed responses.

e using the last selected targetafter all priority peers were exhausted.

ens and `cache_read_input_tokens` were excluded from prompt token counts, causing undercounting
when Anthropic prompt caching was active.

    - Fixed intermittent upstream 401s caused by reused expired Azure managed-identity tokens.

- datakit:
    - Fixed an issue where configuration parsing failed in packaged builds due to a missing jq parser file.
- openid-connect:
    - Fixed an issue where token endpoint responses with `expires_in <= 0` were cached indefinitely when proactive expiry re-check was disabled, causing the plugin to silently forward already-expired tokens to the upstream until a worker/process restart.

Clustering

- Fixed an issue where data planes retried unrecoverable TLS certificate setup failures instead of loading their backup configuration.
- Fixed an issue where pushing a configuration with a datakit plugin whose node `inputs` contained a null value to an older data plane could crash the control plane's config compatibility checker with "attempt to index local 'value' (a userdata value)".

Core

- **router**: Fixed an issue where WebSocket routes (`ws`/`wss`) returned HTTP 404 when using `router_flavor=expressions` due to incorrect protocol matching expressions being generated.

3.14.0.7

Release date 2026/06/23

Bugfix

Core

- Added the missing PDK functions to sandbox environments.
- **debugger**: Fixed an issue where response body content capture consumed streaming response chunks.
- **Debugger**: Fixed an issue where the default key name `apikey` of the `key-auth` and `key-auth-enc` plugins was not redacted when captured by the Debugger. This has been resolved by adding `apikey` to the list of default redacted headers.
- Fixed an issue where a failure in a `dao:*:post` hook during insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.
- Fixed an issue where cluster mutex did not timeout immediately with `no_wait = true` when node level mutex is being held by another worker.
- Fixed an issue where immediate cosocket SSL handshake errors could be reported as subsequent socket write failures.
- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. Now includes a cert fingerprint for proper connection isolation.
- Fixed an issue where the `KONG_ANALYTICS_FLUSH_INTERVAL` environment variable was not correctly applied to the analytics flush interval.

Plugin

- **kafka-upstream, kafka-log, kafka-consume, confluent, confluent-consume**:
    - Fixed Kafka keepalive pools leaking connections across different SASL credentials and TLS verification settings.
- ai-mcp-oauth2:

    hen `client_auth` was set to `private_key_jwt` and `client_jwk` was provided as a JSON string.

    sts lost the original client IP address, causing IP-aware plugins like ip-restriction to reject legitimate

    erver responses were rewritten to empty objects when the plugin re-encoded the body for ACL list filtering.

    asons are not mapped in OpenAI to Anthropic conversion.

    the `top_p`, `top_k`, and `stop` parameters when translating OpenAI-format requests. The driver now also
    hat reject them (Claude Opus 4.7, Opus 4.8, and Fable), preventing upstream 400 errors.

    nstruct upstream AI provider paths.

    pstream AI provider path traversal.

    ve format adapters could cause upstream AI provider path traversal.

    ing_effort` under `reasoning.effort` and did not normalize `reasoning_content` in responses.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where native-format requests using a model alias were incorrectly proxying to upstream model.

- ai-proxy:
  - Fixed an issue where responses with unsupported content types could be returned with an incorrect gzip header.
  - Fixed an issue where the Azure managed identity token was cached indefinitely and never refreshed, causing requests to fail with 401 after the token expired (roughly one hour).
- ai-rate-limiting-advanced:
  - Emitted the documented `X-AI-RateLimit-*` headers on `429` responses from policies that partition by `provider` or `model`.
  - Fixed policy integration with `ai-proxy` for single-target and multi-target AI proxy routes.
- ai-request-transformer:
  - Fixed an issue where unsafe model name could cause upstream AI provider path traversal.
- ai-semantic-response-guard:
  - Fixed an issue where the plugin failed to populate Redis with rule documents due to using worker ID gating instead of worker mutex for vectordb operations.
- datakit:
  - Added Lua control plane compatibility checks to prevent unsupported Datakit configurations from being pushed to older data planes in hybrid mode.
- exit-transformer:
  - Fixed an issue where a response body that was a JSON primitive (e.g. `false`) caused a 500 internal error.
- openid-connect:
  - Fixed an inconsistency where `credential_claim` resolved the same claim name to a different value than `upstream_headers_claims` because it consulted the `id_token` before the `access_token`, while the upstream headers resolver consulted the `access_token` first. With IdPs (like Okta) that set distinct `sub` values in the two tokens, `authenticated_entity.id` did not match the upstream header the plugin set on the same request.
  - Fixed an issue where the `redirect_uri` parameter was sent to the token endpoint for all grant types. It is now included only for the `authorization_code` grant, as required by RFC 6749.
- opentelemetry:
  - Fixed an issue where request-scoped `resource_attributes` could leak across concurrent queued `traces`, `logs`, and access log exports when the plugin was configured globally.
  - Fixed an issue where the plugin crashed when DNS or balancer failures occurred during upstream retries.
- session:
  - Fixed an issue where concurrent authentication requests using Postgres for session storage caused race conditions and session creation collisions.
- tcp-log:
  - Fix the race condition where the balancer `tries` array gets cleared before the asynchronous timer thread can safely encode it.

Clustering

- Fixed an issue where DP syncing failed with large configs. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful full-sync.
- Fixed an issue where hybrid data planes retried unrecoverable CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable reconnect storms.

pending foreign-key references across a failed resume, causing spurious validation failures.

`_ssl_verify_depth` from 1 to 5.

0 to solve an issue when handling protocol upgrade failure.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **ai**:
  - Removed unused redis connection when setting up Redis vectordb.

Core

- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

3.14.0.6

Release date 2026/06/11

Bugfix

Core

- Applied upstream nginx security patches for limiting the maximum number of headers (CVE-2026-49975).

3.14.0.5

Release date 2026/06/04

Feature

Clustering

- Added a safeguard to prevent the CP or DP from sending payloads larger than the configured `cluster_max_payload` size to each other, which could avoid the frequent disconnections and reconnection attempts and improve performance. Emitted a log entry telling the config payload size and the configured `cluster_max_payload` size when this happens.

Bugfix

Plugin

- **ai**:
  - Fixed an issue where non-200 responses from Bedrock embeddings API were not properly handled, leading to incorrect feedback to the client.
- **ai-guardrails**:
  - Fixed an issue where multiple guardrail plugins couldn't function together.
- ai-mcp-proxy:
  - Fixed an issue where converted HTTPS MCP tool calls lost the original forwarded port on the internal TLS Unix socket.
  - Fixed stale tools cache after full-sync reconfiguration (e.g. after a `deck sync`), causing `tools/list` to return empty results.
- ai-proxy-advanced:
  - Fixed an issue where full-sync reconfiguration would reset the balancer state even when plugin configs were unchanged.
  - Fixed an issue where driver transformers returned errors or stream metadata in the wrong return position.
  - Fixed an issue where OpenAI-compatible Anthropic chat requests only preserved the last system message.

asn't used for the Azure provider.

ame in OpenAI streaming responses from Azure.

s forwarded to Databricks upstreams, causing a 400 error since Databricks does not support this field.

s (e.g. `$(uri_captures.*)`) in `config.llm.model.*` fields were not interpolated.

s (e.g. `$(uri_captures.*)`) in `config.llm.model.*` fields were not interpolated.

) were incorrectly converted to objects (`{}`) when using `is_proxy_integration=true` and ansformer plugin.

_proxy_integration` was enabled and the Lambda function returned `null` values in `headers` or d to empty strings.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- forward-proxy:
  - Added the `ca_certificates` parameter for per-plugin TLS verification. This setting allows certificate chain and hostname checks against specified CAs instead of global settings, ensuring explicit trust.
- jwt-signer:
  - Fixed an issue where the plugin failed to load configuration and errored when `null` or `nil` values were passed to `access_token_keyset` or `channel_token_keyset`.
- oas-validation:
  - Fixed an issue where the OAS specification cache failed to distinguish between different values of `api_spec_encoded` for the same specification content.
  - Fixed an issue where path parameters were extracted from the wrong URI segment when `custom_base_path` was used with bare `/{param}` paths.
- openid-connect:
  - Fixed an issue where Redis-backed session storage could fail when Redis connections were routed through a proxy.
- opentelemetry:
  - Fixed an issue where the template `resource_attributes` in the plugin configuration didn't work as expected for metrics.
- request-validator:
  - Added `array_length_compat` config option to allow legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.
- ai-semantic-prompt-guard, **ai-semantic-response-guard**: Fixed an issue where full-sync reconfiguration rebuilt semantic vectorDB indices even when the plugin configuration was unchanged.

Clustering

- Fixed an issue where a config backup exported by a data plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or `oic_issuers`. The export incorrectly included a `ws_id` field for those entities, which the declarative config schema rejects as an unknown field.
- Fixed an issue where incremental DP (sync v2) would not export config to fallback storage when the control plane used periodic polling instead of `notify_new_version`.

Core

- Fixed an issue where Kong Manager users authenticated through IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim changes, or end up with inconsistent mappings during concurrent authentication requests.

Dependency

Core

- Bumped kong-lua-resty-ljsonschema from 1.3.1 to 1.3.3.

Performance

Core

- Improved performance of analytics data collection by introducing a shared payload pool and reducing the default local buffer size.

3.14.0.4

R

F

C

40701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.

ader handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header `Connection: Content-Length` headers on requests with a body.

D

C

: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bumped OpenSSL from 3.5.5 to 3.5.6 to address the issues: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-2026-28390, and CVE-2026-31789.

3.14.0.3

Release date 2026/05/25

Bugfix

Plugin

- **ai-guardrail**:
  - Fixed an issue where latency metrics were missing in the Prometheus output.
- ai-azure-content-safety:
  - Fixed a crash (HTTP 500) that occurred when `azure_use_managed_identity` was enabled.
- ai-mcp-oauth2:
  - Fixed an issue where downstream scope-based ACL evaluation used claims from the inbound token instead of the exchanged token when token exchange was enabled.
- ai-mcp-proxy:
  - Fixed an issue where, since 3.14, call tools converted from API could fail when Kong used a self-signed certificate.
  - Fixed an issue where the tool ID might be repeated.
  - Fixed an issue where raw non-200 upstream responses triggered misleading MCP body parse warnings.
- ai-prompt-decorator:
  - Fixed an issue where the plugin corrupted native-format requests (Gemini, Anthropic, Bedrock, Cohere, Hugging Face) by forwarding an OpenAI-shaped body upstream instead of the original native body shape.
- ai-proxy-advanced:
  - Fixed an issue where GenAI spans were not finished sometimes.
  - Fixed an issue where we didn't support Azure GA version of Realtime API.
  - Fixed an issue where OpenAI-format chat reasoning requests were not mapped consistently across Anthropic, Gemini, Bedrock and more providers.
  - Fixed an issue where system prompt of Claude Code was truncated when doing non-passthrough proxy.
  - Fixed an issue where we didn't use Hugging Face's token usage when it was available.
  - Fixed an issue where we never stored the response model in the streaming path, causing it to always fall back to the request model.
  - Fixed an issue where DashScope replied to inference requests with a 500.
- ai-sanitizer:
  - Fixed an issue where the AI sanitizer plugin could not parse the request body correctly.

Core

- Fixed an issue where anonymous reports omitted model usage when one provider used multiple models.
- tatables from previous TLS connections, leading to cross-request context leakage and increased memory

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

AI Response Transformer plugins did not work properly with the Ollama provider.

alid legacy ACL group credentials were rejected with 403 when `include_consumer_groups` was enabled and no enterprise consumer groups were assigned.

- ai-mcp-oauth2:
  - Fixed an issue where token exchange actor tokens were not sourced correctly from `token_exchange.request`, ensuring correct forwarding of actor tokens configured in headers or plugin config.
  - Fixed an issue where `token_exchange.cache.enabled = false` was ignored and exchanged tokens were still cached because the cache toggle incorrectly read `token_exchange.cache.ttl` instead of `token_exchange.cache.enabled`.
- ai-mcp-proxy:
  - Fixed an issue where the `WWW-Authenticate: Bearer error="insufficient_scope"` header was not emitted.
- ai-proxy-advanced:
  - Fixed an issue where observability errors occurred when `gpt-image-1.5` returned image token usage details.
  - Fixed an issue where Azure OpenAI authentication failed during load balancer failover due to token refresh attempts in `balancer_by_lua`. Tokens are now pre-fetched in the `access` phase.
  - Fixed an issue where we treated plaintext Anthropic response as gzipped.
  - Fixed an issue where we didn't correctly ungzip the streaming response.
  - Fixed an issue where Claude Code (or CLI) statistics were missing when using an Anthropic model via the Bedrock provider.
- graphql-rate-limiting-advanced:
  - Fixed an issue where the GraphQL Rate Limiting Advanced plugin's API threw 500 errors when handling HTTP PUT requests.
- jwt:
  - Fixed an issue where malformed JWT tokens (e.g. tokens without dots) caused a 500 Internal Server Error instead of returning 401 Unauthorized.
- kafka-log:
  - Fixed an issue where cached producers could cause memory leaks.
- opa:
  - Fixed json body processing to recognize `application/json` content-type with charset, ensuring proper handling of JSON request bodies.
- openid-connect:
  - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.
- service-protection:
  - Fixed an issue where the Service Protection plugin could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after Rate Limiting Advanced and only evaluates requests that pass rate limiting, by updating its priority from 915 to 901.

Core

- **Debugger**: Fixed an issue where Redis command arguments were incorrectly included in Redis spans.
- **Debugger**: Fixed an issue where the time spent during the TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side latency.
- **Debugger**: Fixed an issue where the Konnect Debugger concatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g. `unix:/<prefix>/sockets/am:tls`).
- Fixed a bug where tables in `strict` and `lax` `untrusted_lua` modes were not properly protected against modification.

h the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` is
dy messages for 501 and 505 status codes.

ol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497
ted prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from
l added `error_page` 497 routing through `kong_error_handler`.

C

n and stop further sync attempts.

mers when downgrading sync protocol to v1, which could cause excessive load on the CP and lead to

lling pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

K

he service entity form.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Dependency

Core

- Bumped kong-gql from 0.2.3 to 0.2.4 to fix the null value issue in the GraphQL query syntax that caused a parsed AST error in DeGraphQL plugin.
- Bumped kong-pgmoon from 2.3.3.0 to 2.3.4.0, which adds SHA-384 support for SCRAM channel binding during PostgreSQL SSL authentication.
- Bumped lua-resty-healthcheck from 3.1.1 to 3.1.2.

3.14.0.1

Release date 2026/04/10

Bugfix

Plugin

- openid-connect:
  - Fixed an issue where nested claim paths were not resolved correctly.

3.14.0.0

Release date 2026/04/07

Breaking Change

Core

- **Configuration**: The `tls_certificate_verify` option is now enabled by default, preventing Kong from bypassing certificate verification on any secure connection.
- **event hooks**: Sign event hook calls with HMAC-SHA256 instead of HMAC-SHA1.
- **oauth2**: the option `ssl_verify`, to verify the certificate presented by the IdP when using HTTPS, is now on by default.
- **Redis Partials**: changed the default value for the `ssl_verify` from `off` to `on`.
- **Routes**: Change default protocols from `http, https` to `https` for routes entities.

Plugin

- **key-auth, key-auth-enc, basic-auth, hmac-auth, ldap-auth, oauth2, oauth2-introspection, vault-auth, ldap-auth-advanced**:
  - Changed the default value of `hide_credentials` from `false` to `true` to enhance security. Users should verify their configurations if relying on the previous default.
- jwt-signer:
  - calls to the `/rotate` endpoint now enable certificate verification by default when connecting to the configured endpoint to rotate the keys.

Deprecation

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

recated the `llm_format` configuration field.

etadata endpoints via `gcp_oauth_token_url` and `gcp_metadata_url` configuration options in the `auth` ed network environments or with custom GCP endpoints.

cify model options separately from the main configuration. Metadata options are merged into the model

- **solace-upstream, solace-log, solace-consume**:
  - Added support for the global `tls_certificate_verify` option. When enabled, plugins using secure protocols (tcps://, wss://, https://) must have `ssl_validate_certificate` set to `true`.
- **vectordb**:
  - Added Valkey support for vector database operations, providing an alternative to Redis Stack for vector search capabilities.
  - The option `ssl_verify`, which control SSL certificate verification for the pgvector database, is now enabled by default.
- acl:
  - Added WebSocket protocol support (`ws`/`wss`). The plugin now enforces allow/deny group rules during the WebSocket handshake phase.
- acme:
  - Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `storage_config.redis.tls_verify` setting cannot be disabled.
- ai-a2a-proxy:
  - Added A2A analytics support for Konnect, enabling visibility into A2A protocol operations including method calls, latencies, and task states.
  - Added new plugin for Google A2A (Agent-to-Agent) protocol support, enabled AI agent interoperability with JSON-RPC and REST bindings, built-in observability.
  - Added Prometheus and OpenTelemetry observability with A2A-specific metrics and tracing spans.
- ai-aws-guardrails:
  - the `ssl_verify` option to validate TLS certificate when connecting to the bedrock service is now enabled by default.
- ai-azure-content-safety:
  - the `ssl_verify` option to verify the certificate presented by the Azure Content Safety service when using HTTPS is now enabled by default.
- ai-custom-guardrail:
  - Added a new custom guardrail plugin which enables integration with 3rd party guardrail service.
- ai-mcp-oauth2:
  - Added support for passing tokens upstream.
  - Added support for multiple token validation methods.
  - Added support for mapping claim to authenticated credential.
  - Added Token Exchange support to swap JWT tokens before accessing MCP Server.
  - Added `upstream_headers` field for mapping token claims to upstream headers using path-based access. Mutually exclusive with `claim_to_header`.
  - Made the client_id field not required when client_auth is set to something other than client_secret_basic or client_secret_post
  - Added support mapping claims in token to `consumer` and `consumer_groups`.
- ai-mcp-proxy:
  - Added support for session sharing and active notice when doing MCP conversion.
  - Added support for ACL with OAuth claim.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

deny rules, requires allow rules to pass when present, and resolves consumer/model values once per

ltimodal embeddings. Introduced new `llm/v1/multimodal-embeddings` route type.

odels on bedrock.

models in same AI Proxy Advanced plugin.

ll Ollama-hosted models.

tion request to cohere provider.

- Added native Anthropic streaming mode support, enabling direct passthrough of Anthropic SSE events with proper message reordering for tool calls.
- Added support for batch API in openai-compatible mode when providers are anthropic, bedrock, gemini and gemini vertex.

- ai-rate-limiting-advanced:
  - Added a policy-based rate limiting mode with multi-dimensional match conditions and request counting strategy.
  - Added prompt token estimation for `prompt_tokens`, `total_tokens`, and `cost` strategies in policies mode, enabling early request rejection before the upstream responds.
- aws-lambda:
  - Added `ssl_verify` config field to control SSL certificate verification when connecting to AWS APIs.
- azure-functions:
  - the `https_verify` option to validate the Azure Functions server certificate is now enabled by default.
- basic-auth:
  - Uses SHA256 by default even in non-FIPS mode instead of SHA1.
- confluent-consume:
  - the `security.ssl_verify` option to validate the server certificate is now enabled by default.
- confluent:
  - the `security.ssl_verify` option is now enabled by default.
- datakit:
  - Added JWT nodes: `jwt_decode` for decoding tokens without verification, `jwt_verify` for signature verification and claim validation using JWKS, and `jwt_sign` for creating and signing JWTs.
  - Added more implicit node outputs: * `request.path` * `request.raw_path` * `request.port` * `request.method` * `request.scheme` * `request.version` * `service_response.status` * service_response.raw_body * call.raw_body
  - Consumer can now be set via property node.
  - the call node's `ssl_verify` option to verify the TLS certificate when making HTTPS requests is now enabled by default.
- forward-proxy:
  - `https_verify` is now enabled by default. This option verifies the upstream server certificate presented after the plugin establishes a `CONNECT` tunnel through the proxy.
- header-cert-auth:
  - the `ssl_verify` option to switch verification of the certificate presented by the server of the OCSP responder's URL and by the server of the CRL Distribution Point is now enabled by default.
- hmac-auth:
  - Add support for HMAC-SHA224, and removes HMAC-SHA1 from the default set of algorithms (still supports HMAC-SHA1 when configured).
- http-log:
  - the `ssl_verify` option to validate the `http_endpoint` server certificate is now enabled by default.
- json-threat-protection:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

figuration to allow non-JSON requests to bypass all checks.

y and `channel_token_endpoints_ssl_verify`, which control SSL certificate verification for signature nabled by default.

SSL certificate verification for Kafka endpoints, are now enabled by default.

the verification of the certificate presented by the server, is now enabled by default.

P server certificate is now enabled by default.

- ldap-auth:
  - the `verify_ldap_host` option to validate the LDAP server certificate is now enabled by default.
- metering-and-billing:
  - Added a new plugin for metering and billing.
- mtls-auth:
  - Added WebSocket protocol support. This authentication plugin can now protect WebSocket connections (`ws` and `wss` protocols) in addition to HTTP/HTTPS. In hybrid mode, using `ws`/`wss` with this plugin requires data planes running DP >= 3.14, or explicit removal of `ws`/`wss` protocols from configurations targeting older data planes.
  - the `ssl_verify` option to validate verification of the certificate presented by the server of the OCSP responder's URL and by the server of the CRL Distribution Point is now enabled by default.
- oauth2-introspection:
  - The authenticated token is now stored in the request context via `kong.client.set_token`.
- oauth2:
  - The authenticated token is now stored in the request context via `kong.client.set_token`.
  - Uses SHA256 for access token cache key instead of SHA1.
- openid-connect:
  - Added `consumer_claims` to support mapping consumers from multiple claim paths, while retaining and supporting the existing `consumer_claim` as a shorthand configuration.
  - Added `jwks_endpoint` configuration to override the discovered `jwks_uri` when a custom JWKS endpoint is required.
  - Added support for token exchange from multiple IdPs based on RFC8693.
  - Added `upstream_headers` and `downstream_headers` schema fields that support nested claims.
  - Added WebSocket protocol support. This authentication plugin can now protect WebSocket connections (`ws` and `wss` protocols) in addition to HTTP/HTTPS. In hybrid mode, WebSocket protocol support for this plugin requires data planes running Kong Gateway 3.14 or later; for older data planes, ensure routes do not use `ws`/`wss` protocols, as their propagation will be blocked.
  - Marked fields `issuer`, `extra_jwks_uris`, and `introspection_endpoint` are now supported as vault references.
  - Marked `issuers_allowed` as a referenceable field.
  - The authenticated token is now stored in the request context via `kong.client.set_token`.
  - the `ssl_verify`(for verifying identity provider server certificate) and the `session_memcached_ssl_verify`(for verifying session storing memcached server certificate) option are now enabled by default as well.
- opentelemetry:
  - Add Konnect Cluster ID and the optional Cluster Member ID attributes to OpenTelemetry metrics in Konnect mode.
  - Added support for AI/MCP metrics
  - Added support for AI/MCP access log
  - Added support for customizing access logs via Lua code snippets. The previous `access_logs_endpoint` schema field has been deprecated in favor of a new `access_logs.endpoint` field, which can be used alongside the `access_logs.custom_attributes_by_lua` field to specify Lua code snippets for custom attributes.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ster Member ID to access logs.

try plugin for enhanced observability.

ich controls the verification of the certificate presented by the server, is now enabled by default.

equest-header fallback) and `forward_body_raw_only` for raw payload passthrough; headers can be

- **active-tracing**: Added auto-decompression for gzipped bodies when capturing payload
- Added request and response content evolution support to the debugger session. This allows the debugger to capture and compare request and response contents across different phases of the request lifecycle, providing insights into how the contents change as they flow through Kong Gateway.
- Added support for PostgreSQL OAuth/OAUTHBEARER SASL authentication. Kong can now authenticate with PostgreSQL using OAuth 2.0 bearer tokens obtained via client credentials grant from an identity provider. This requires PostgreSQL 18+ with OAUTHBEARER support and the updated pgmoon library with OAUTHBEARER SASL implementation.
- Added support for sampling rate in sessions of active tracing
- **Conditional Plugin Execution**: Added support for conditional plugin execution through a new `condition` field in the plugin configuration. Plugins can now be conditionally skipped at runtime based on request attributes such as HTTP method, path, headers, query parameters, consumer, route, service, and network information. The condition expression uses the ATC expression language and is evaluated once during the collecting phase; skipped plugins are excluded from all downstream phases automatically.
- **core**: Added support for decoding JSON null values in objects as Lua `nil` via `cjson.decode_null_object_value_as_nil` (and exposed as `kong.tools.cjson.decode_null_object_value_as_nil`).
- **debugger**: Add CPU profiling support for debugger to Kong.
- **debugger**: Added sanitization support for payload using configurable redaction rules.
- **debugger**: Added status code and status message to plugin span attributes when using kong.response to exit.
- **DP Resilience**: Added support for Azure Blob Storage in config sync backup strategy.
- **event-hooks**: The options `ssl_verify` for the webhook type handler are now enabled by default.
- **Hashicorp Vault**: Added AWS auth method authentication to the HashiCorp Vault backend.
- **Hashicorp Vault**: Added Azure authentication to the HashiCorp Vault backend.
- **Hashicorp Vault**: Added GCP auth method authentication to the HashiCorp Vault backend.
- **Hashicorp Vault**: Added `ssl_verify` config field to control SSL certificate verification when connecting to Hashicorp Vault servers.
- Kong will validate the database connection configuration at startup and will not start if errors are detected.
- **sandbox**: Added `strict` and `lax` modes for `untrusted_lua`. The default mode changed from `sandbox` to `strict`.
- Upgraded lua-resty-aws library to the 1.7.2 version.

Configuration

- Added a new configuration option `schema_alias_conflict_mode` that controls the behavior when a deprecated alias field and its canonical replacement field are both present with mismatched values. The default value `error` preserves the existing strict validation behavior. Setting it to `warn` accepts the configuration with a warning log and always uses the canonical field value, allowing upgrades to proceed without requiring all legacy plugin configurations to be corrected first.
- Added `lru_cache_size` to allow configuration of in-memory LRU cache size.
- **core**: Add `lua_gc_tuning` option to control garbage collection parameters on Control Plane.

PDK

- Added `kong.client.get_jwt_token_payload()` to retrieve a decoded JWT token payload.
- Added `kong.client.get_jwt_token_header()` to retrieve a decoded JWT token header.
- Added `kong.client.get_token()` function to retrieve a shared authenticated token.

lenticated token with other plugins.

P

cluding upstream, third-party I/O, and client in the HTTP context for Prometheus, Analytics, and logs.

P

P

validate message keys when using Confluent Schema Registry.

h incorrect `offsetDelta` in the record package.

on failed silently, returning raw string data instead of deserialized objects.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **kafka-consume, confluent-consume**:
  - Increased the minimum API version of FetchRequest from 0 to 4 for Kafka 3.x. For Kafka 4.0 it is already using version 4.
- **kong.service.request.clear_query_arg**:
  - Fixed an issue where an extra = was appended to empty query arguments.
- **llm**:
  - Fixed an issue where we could not specify the correct dimensions for Gemini embeddings model provider.
  - Removed ineffective `ssl_cafile` parameter from HTTP requests. This parameter is not supported by lua-resty-http and was silently ignored. TLS verification relies on the global `lua_ssl_trusted_certificate` nginx directive.
- **solace-upstream, solace-log, solace-consume**:
  - Fixed an issue where enabling `ssl_validate_certificate` would fail because `SESSION_SSL_TRUST_STORE_DIR` was not set.
- acl:
  - Fixed an issue where the ACL plugin did not check consumer groups from all sources, causing it to incorrectly deny access when consumer groups were set by other plugins like ACE in `ngx.ctx.authenticated_consumer_groups`.
- ai-aws-guardrails:
  - Allowed streaming of responses when `guarding_mode` was set to `INPUT` with `allow_masking` enabled.
- ai-gcp-model-armor:
  - By default now blocks unrecognized/new filters added to GCP Model Armor.
- ai-mcp-oauth2:
  - Fixed an issue where we didn't clear header for absent claim.
- ai-mcp-proxy:
  - Fixed an issue where ACL denial caused parsing body failure.
  - Fixed an issue where the ACL metric label value was incorrect.
  - Fixed an issue where `tools/list` could expose tools to consumers denied by tool ACL in listener mode.
  - Fixed an issue where we didn't keep the port after MCP conversion.
  - Fixed an issue where we have not reported metrics to konnect since 3.13.
  - Fixed an issue where PATCHing the tool caused an error.
  - Fixed an issue where the latest 2025-11-25 spec defines the Authorization error codes which are different from ours.
- ai-proxy-advanced:
  - Fixed an issue where metrics were missing and v2 API was unavailable in Cohere native format.
  - Fixed an issue where id/created fields were missing in multiple drivers.
  - Fixed an issue where gzip streaming response in OpenAI format was broken.
  - Fixed an issue where content-type header, from non-200 response of streaming request, was overridden.
  - Fixed an issue where GCP tokens would not be proactively refreshed before expiration.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

re incorrect according to https://opentelemetry.io/docs/specs/semconv/gen-ai/gen-ai-spans.

riable endpoints like `$(headers.x_gcp_endpoint)`. These warnings appeared on every configuration ed until request time.

lso Llama2 and Mistral) providers.

`le_id}/content` batch output analytics.

`ponses` in streaming mode.

istral) providers.

tream returned `choices: null` (JSON null). The `header_filter` phase would crash with "attempt to get `ngx.null` is truthy in Lua.

routes did not collect metrics and logging.

models hosted on Vertex AI (claude-, *mistral*-, jamba-*) that reported inflated totals because incremental counts were treated as cumulative.

- ai-rate-limiting-advanced:
  - Extended deferred rate limiting callback to support model-partitioned policies and cost strategy.
  - Fixed an issue where the plugin did not validate whether the configured `dictionary_name` exists before use, which could lead to runtime errors.
- ai-sanitizer:
  - fixed an issue where the sanitizer plugin always sanitize the entire request body, which was fixed with latest message only.
- ai-semantic-prompt-guard:
  - Fixed an issue where a vectorDB error occurred during initialization or config reload.
- aws-lambda:
  - Fixed an issue where headers from Lambda proxy responses were not merged correctly, causing case-insensitive header deduplication to fail. User-supplied headers now properly overwrite existing headers regardless of casing.
  - Fixed an issue where Lambda responses with empty or null body caused 500 errors when `is_proxy_integration` is enabled.
- basic-auth:
  - Brute Force Protection: when using redis strategy and the connection fails, the plugin will now fall back to using the memory strategy.
- datakit:
  - Fixed an issue where datakit cache node instantiate proxy-cache-advanced redis module at config time causing dependency issue on Koko.
  - Fixed type system bug to allow defining maps with string values for vault node.
  - Fixed xml_to_json node to recognize `text/xml` content-type in addition to `application/xml` for XML request body processing, ensuring RFC 7303 compliance.
- degraphql:
  - Fixed an issue where the plugin failed to parse GraphQL variables with nested array types like `[String]!`. The error `attempt to index field 'name' (a nil value)` occurred because the type parser did not correctly handle deeply nested type structures such as `nonNullType(listType(namedType))`.
- graphql-rate-limiting-advanced:
  - Fixed an issue where the plugin could fail to connect to the upstream server to introspect the upstream schema.
- injection-protection:
  - Fixed an issue where the plugin was too sensitive to certain injection patterns.
- ldap-auth-advanced:
  - Fixed a security issue.
- ldap-auth:
  - Fixed a security issue.
- mtls-auth:
  - Fixed an issue where client certificates with Subject Alternative Name (SAN) DirectoryName extensions caused authentication errors. Added a new `san_dirname_matcher` configuration option to specify which Distinguished Name attributes from the SAN DirectoryName extension should be used for consumer lookup.
- oas-validation:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

used a runtime crash when `collect_all_errors` was enabled.

s were not properly handled.

nstraints defined in a Swagger 2.0 parameter schema were not enforced.

count for the `collect_all_errors` setting.

not properly disabled when `validate_response_body` was set to `false`.

ion/json content-type with charset, ensuring proper handling of JSON request bodies.

were used for Redis Sentinel and no authentication supported for Redis, causing `NOAUTH Authentication`

be forwarded to upstream services when the corresponding JWT claim was missing from the token. The `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or

- opentelemetry:
  - Fixed an issue where invalid `x-b3-flags` header values caused 500 responses.
  - Fixed an issue where the `kong.upstream.status_code` field of access logs was incorrectly populated with string value instead of integer value when there was one or more retries to the upstream.
  - Fixed an issue where the `http.response.status_code` attribute of the `http.server.request.count` metric was incorrectly set to `kong.response.status_code`. Now, the attribute name follows the OTel specification.
  - Fixed an issue where valid response codes from vendor endpoints were incorrectly rejected when exporting `traces`, `logs` or `metrics`.
- request-callout:
  - Fixed keepalive support in callout HTTP requests to improve connection reuse and performance.
- request-transformer:
  - Fixed an issue where the `add.headers` operation lowercased the header name, causing the configured letter case to be lost when adding new headers to the request.

Core

- Apply Nginx patch for a man-in-the-middle upstream response injection (CVE-2026-1642)
- **core**: Resolved an issue where HTTPS was incorrectly detected if multiple `X-Forwarded-Proto` values were provided.
- Fixed an issue where context sensitive plugins would not execute when dynamic ordering is enabled in the plugins iterator.
- Fixed an issue where DP's proxy listener could not be verified because of it was disabled in CP.
- Fixed an issue where functions in required sandbox modules could not be called in sandboxed Lua code due to the `__call` metamethod not being properly handled.
- Fixed an issue where GCP Vault authentication method order could not be set by the `GCP_AUTH_METHOD_ORDER` environment variable.
- Fixed an issue where in the postinstall script that changed /etc/kong/kong.logrotate's ownership and permissions; the script now restores ownership to root:root and sets permissions to 0644.
- Fixed an issue where `kong health` command would fail to run when kong configuration contains vault references.
- Fixed an issue where PDK logs were not captured by observability when your gateway log level was higher than the log entry level.
- Fixed an issue where the Active Tracing would report incorrect `latencies.client` values due to not capturing the access end time properly. This fix ensures that the access end timestamp is always recorded regardless of Active Tracing being active or not.
- Fixed an issue where the `network.peer.name` attribute in OpenTelemetry traces showed the same hostname for all retry attempts instead of capturing the specific hostname for each attempt.
- Fixed an issue where the Redis cloud authentication feature cannot fetch a correct credential for connecting to Azure Redis, especially when using Azure Workload Identity.
- Fixed an issue where the request and response contents (headers and body) of external HTTP or Redis calls were not being recorded in the Content Evolution Capture.
- Fixed an issue where the tracing PDK was missing in the Sandbox environment.
- Fixed an issue where, when the global tls_certificate_verify option is enabled, service entities with an unset tls_verify field would fail validation. The tls_verify field now defaults to the proxy_ssl_verify nginx setting when not explicitly set.
- Fixed AWS STS throttling during Redis IAM authentication via token caching.
- Fixed external plugin server to properly return 500 error responses when RPC calls fail, instead of silently ignoring the failures.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ged the secret to an empty string temporarily after changed vault config.

vices did not fall back to Nginx's `proxy_ssl_verify` when `tls_verify` was left unset.

C

d in the clustering RPC procedure, ensuring safer SQL operations.

properly clean up when config changed in CP/DP mode.

use the Application Default Credentials method order when the environment variable

C

y via `kong reload` did not take effect because the derived `proxy_ssl_verify` directive persisted in

Dependency

Core

- Bumped lua-resty-gcp from 0.0.16 to 0.0.19 to address GCP access token handling improvements.
- Bumped OpenSSL from 3.4.3 to 3.5.5 to address the issue: CVE-2025-15467.

Performance

Plugin

- ai-mcp-proxy:
  - reduced tool cache build time and avoided long blocking.
- ai-proxy-advanced:
  - decoded OpenAI response only when necessary.

Core

- **core**: Reduced transient table churn and associated GC pressure by introducing preallocated table pools for hot-path table allocations.

Admin API

- Optimized `GET /kong` endpoint to use `SELECT DISTINCT` for populating `enabled_in_cluster` instead of iterating all plugin rows.

3.13.0.10

Release date 2026/08/20

Bugfix

Plugin

- ai-proxy-advanced:
  - Fixed a bug where a response from Anthropic could fail to decode when the client's request advertised Brotli support, since Anthropic's edge may then return a Brotli-encoded body that Kong's response decompression – which only understands gzip – can't inflate. Kong now normalizes the outgoing `Accept-Encoding` request header before proxying to any LLM provider: `gzip, identity` when the client accepts gzip, and `identity` otherwise, so a client that cannot inflate gzip is never handed a compressed body on the paths that pass the upstream response through untouched.
- datakit:
  - Fixed an issue where the `response` node set `Content-Type: application/octet-stream` for `json_to_xml` output. It now defaults to `application/xml` when no explicit `Content-Type` is provided.
- jwt-signer:
  - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.
- mocking:
  - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oauth2-introspection:

expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false`

was sent to the token endpoint for all grant types. It is now included only for the `authorization_code`

when it is set, because it turns off JWS signature verification for the listed grants.

`expiry` metric reused the same datapoint timestamp across exports, causing monitoring systems to treat the

ure Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that
s, so a signature can no longer be bound to a different element than the one that provides the identity. A
me-document fragment is now rejected.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a crash that occurred when a plugin set `schema_registry.confluent.authentication.mode` to `oauth2` but omitted the `oauth2_client` record. Previously this returned a 500 from the Admin API during validation, and once saved it also crashed at runtime while fetching the schema registry OAuth2 token. The absent `oauth2_client` is now handled and falls back to the default authentication method, so a configuration without `oauth2_client` works end-to-end.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed an issue where OAuth2 authentication against Confluent Cloud Schema Registry failed because the client did not send the required `target-sr-cluster` header.

PDK

- Fixed an issue on HTTP/2 connections where the request method and query were reported incorrectly, or could raise an error, when a request hit the error handler such as a 502 from an upstream connection reset.

Core

- Fixed an issue where a failed balancer creation leaked the per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.
- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.
- Fixed an issue where `/status/ready` could report a data plane ready before the new router and plugins iterator became active. Kong now reports readiness only after plugin configuration completes and the live objects are installed.
- Fixed an issue where upstream failover could return intermittent 503 responses while a failover target remained healthy.
- **Konnect Only**. **Debugger**: Fixed an issue where deep copy of the worker logs stack overflowed by replacing `deep_copy` with `cycle_aware_deep_copy`.
- On shell-free (distroless) images, the pluginserver query and start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with a clear error on an unterminated quote or `${`, instead of passing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

Clustering

- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.
- Improved RPC connection stability during long-running calls: a data plane now cancels a timed-out RPC so a late response cannot disrupt later calls, and the connection is kept alive (within a bound) while calls are still in flight instead of being dropped as a missed heartbeat.

CLI Command

- Hardened stale nginx PID detection to survive PID reuse. On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading `/proc/<pid>/cmdline`, so a reused PID from an ungraceful exit no longer blocks a restart.

Kong Manager

- Fixed an issue where the Export button on the Vitals Reports page did not download the report CSV.

Performance

Clustering

- Improved data plane stability when applying a large configuration sync from the control plane, reducing the chance of a worker restart while the update is being processed.
- Optimized performance of clustering payload handling for version compatibility. This reduces CPU usage and transient memory pressure.

A

unter=true. Entity counts for the whole page are now retrieved with one grouped query per entity type
space, while keeping the counts exact.

C

otifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in

3

R

E

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Applied upstream nginx security patches for CVE-2026-42533.
- Applied upstream nginx security patches for CVE-2026-56434.
- Applied upstream nginx security patches for CVE-2026-60005.
- Applied upstream nginx security patches for CVE-2026-42055 and CVE-2026-48142.

Plugin

- saml:
  - Security: Fixed CVE-2026-14917 in the SAML plugin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly (and logging a warning if misconfigured), preventing a crafted unsigned SAML response from silently bypassing authentication under configurations that set `validate_assertion_signature` to false.

Dependency

Core

- Bumped jq from 1.8.1 to 1.8.2.

3.13.0.8

Release date 2026/07/27

Feature

Core

- Added a Distroless Docker image variant for Kong API Gateway. The Distroless image ships only the Kong binary and its runtime dependencies, with no package managers, no shells or any other programs you would expect to find in a standard Linux distribution.

Bugfix

Plugin

- ai-mcp-oauth2:
  - Fixed an issue where token-related 401 responses omitted the protected resource metadata URL when a token was given.
- ai-mcp-proxy:
  - Fixed an issue where non-managed MCP listeners advertised unsupported SSE streams.
  - Fixed an issue where the ACL decision cache has collision.
- ai-proxy-advanced:
  - Fixed an issue where unsafe model names could construct upstream AI provider paths.
  - Fixed an issue where the configured auth header (`api-key` or `x-api-key`, for example) leaked the provider credential into access logs instead of being redacted.
  - Fixed an issue where unsafe batch IDs could cause upstream AI provider path traversal.
  - format adaptors could cause upstream AI provider path traversal.
  - the `top_p`, `top_k`, and `stop` parameters when translating OpenAI-format requests. The driver now also hat reject them (Claude Opus 4.7, Opus 4.8, and Fable), preventing upstream 400 errors.
  - ent style.
  - (`cachedContent` or `safetySettings`, for example) and Vertex AI-only fields (`labels`, for example)
  - use upstream AI provider path traversal.
  - ude the plugin name, making it difficult to identify the source of blocked requests.
  - failed to distinguish between different values of `api_spec_encoded` for the same specification content.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- openid-connect:
  - Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.
  - Fixed an issue where Consumer Groups couldn't be resolved when the groups claim was only available from the user info endpoint, which resulted in a `500` error.

Clustering

- Fixed an issue where data planes retried unrecoverable TLS certificate setup failures instead of loading their backup configuration.
- Fixed an issue where heavy packet loss may prevent the full sync from resuming.

Core

- **Konnect Only**. **Debugger**: Fixed an error that could occur under sustained HTTPS traffic.
- **router**: Fixed an issue where WebSocket Routes (`ws`/`wss`) returned HTTP 404 when using `router_flavor=expressions` due to incorrect protocol matching expressions being generated.

Dependency

Core

- Bumped kong-redis-cluster from 1.5.7 to 1.6.0.

Performance

Core

- Improved router update performance during incremental sync by updating service data in place instead of triggering a full router rebuild when only service attributes change.

3.13.0.7

Release date 2026/06/30

Feature

Clustering

- Added a safeguard to prevent the CP or DP from sending payloads larger than the configured `cluster_max_payload` size to each other, which could avoid the frequent disconnections and reconnection attempts and improve performance. Emitted a log entry telling the config payload size and the configured `cluster_max_payload` size when this happens.

Bugfix

Plugin

- **kafka-upstream, kafka-log, kafka-consume, confluent, confluent-consume**:
  - Fixed Kafka keepalive pools leaking connections across different SASL credentials and TLS verification settings.
- ai-azure-content-safety:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ed_identity is enabled.

e.g. deck sync), causing tools/list to return empty results.

when gpt-image-1.5 returned image token usage details.

oken was cached indefinitely and never refreshed, causing requests to fail with 401 after the token expired

s (e.g. $(uri_captures.*)) in config.llm.model.* fields were not interpolated.

  - Fixed an issue where runtime template placeholders (e.g. `$(uri_captures.*)`) in `config.llm.model.*` fields were not interpolated.

- aws-lambda:
  - Fixed an issue where empty JSON array bodies (`[]`) were incorrectly converted to objects (`{}`) when using `is_proxy_integration=true` and `empty_arrays_mode=correct` alongside the exit-transformer plugin.
  - Fixed an issue where a 500 error occurred when `is_proxy_integration` was enabled and the Lambda function returned `null` values in `headers` or `multiValueHeaders`. Null values are now converted to empty strings.
- datakit:
  - Added Lua control plane compatibility checks to prevent unsupported Datakit configurations from being pushed to older data planes in hybrid mode.
- exit-transformer:
  - Fixed an issue where a response body that was a JSON primitive (e.g. `false`) caused a 500 internal error.
- forward-proxy:
  - Added `ca_certificates` option for per-plugin TLS verification. It allows certificate chain and hostname checks against specified CAs instead of global settings, ensuring explicit trust.
- oas-validation:
  - Fixed an issue where path parameters were extracted from the wrong URI segment when `custom_base_path` was used with bare `/{param}` paths.
- opa:
  - Fixed json body processing to recognize `application/json` content-type with charset, ensuring proper handling of JSON request bodies.
- openid-connect:
  - Fixed an inconsistency where `credential_claim` resolved the same claim name to a different value than `upstream_headers_claims` because it consulted the id_token before the access_token, while the upstream headers resolver consulted the access_token first. With IdPs (e.g. Okta) that set distinct `sub` values in the two tokens, `authenticated_entity.id` did not match the upstream header the plugin set on the same request.
  - Fixed an issue where Redis-backed session storage could fail when Redis connections were routed through a proxy.
- opentelemetry:
  - Fixed an issue where template 'resource_attributes' in configuration of the plugin could not work as expected for metrics.
  - Fixed an issue where the plugin crashed when DNS or balancer failures occurred during upstream retries.
- request-validator:
  - Added `array_length_compat` config option to allow legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.
- session:
  - Fixed an issue where when Postgres strategy is used for session storage, concurrent authentication requests could result in race condition and session creation collisions.

Clustering

- **Clustering**: The `/status/sync` endpoint now returns an empty stat object instead of null before the first sync has run, keeping the response shape consistent.
- Fixed an issue where a config backup exported by a data plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or `oic_issuers`. The export incorrectly included a `ws_id` field for those entities, which the declarative config schema rejects as an unknown field.
- Fixed an issue where DP syncing failed with large configs. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful full-sync.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

able CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable

t export config to fallback storage when the control plane uses periodic polling instead of

pending foreign-key references across a failed resume, causing spurious validation failures.

use the Application Default Credentials method order when the environment variable

C

ikey of the `key-auth` and `key-auth-enc` plugins was not redacted when captured by the Debugger. This redacted headers.

ng insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.

diately with `no_wait = true` when node level mutex is being held by another worker.

errors could be reported as subsequent socket write failures.

- Fixed an issue where Kong Manager users authenticated through IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim changes, or end up with inconsistent mappings during concurrent authentication requests.
- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. Now includes a cert fingerprint for proper connection isolation.
- Fixed an issue where the KONG_ANALYTICS_FLUSH_INTERVAL environment variable was not correctly applied to the analytics flush interval.

Configuration

- Increased the default `proxy_ssl_verify_depth` and `lua_ssl_verify_depth` from `1` to `5`.

Default

- **jwt-signer**: Fixed an issue where the plugin failed to load configuration and errored when `null` or `nil` values were passed to `access_token_keyset` or `channel_token_keyset`.

Dependency

Core

- Bumped kong-lua-resty-ljsonschema from 1.3.1 to 1.3.3.
- Bumped lua-resty-websocket from version 0.4.0.2 to 0.5.0 to solve an issue when handling protocol upgrade failure.
- Bumped OpenSSL from 3.5.6 to 3.5.7.

Performance

Core

- Improved performance of analytics data collection by introducing a shared payload pool and reducing the default local buffer size.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

3.13.0.6

Release date 2026/06/11

Bugfix

Core

- Applied upstream nginx security patches for limiting the maximum number of headers (CVE-2026-49975).

3.13.0.5

Release date 2026/05/28

Bugfix

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

0701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.

ader handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header
`Connection: Content-Length` headers on requests with a body.

: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-

es: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-

Bugfix

Plugin

- ai-mcp-proxy:
  - Fixed an issue where we didn't keep the port after MCP conversion.
  - Fixed an issue where ACL denial caused a body parsing failure.
- ai-prompt-decorator:
  - Fixed an issue where the plugin corrupted native-format requests (gemini, anthropic, bedrock, cohere, huggingface) by forwarding an OpenAI-shaped body upstream instead of the original native body shape.
- ai-proxy-advanced:
  - Fixed excessive DNS warning logs for template variable endpoints like `$(headers.x_gcp_endpoint)`. These warnings appeared on every configuration change because template variables cannot be resolved until request time.
- jwt:
  - Fixed an issue where malformed JWT tokens (e.g. tokens without dots) caused a 500 Internal Server Error instead of returning 401 Unauthorized.
- request-transformer:
  - Fixed an issue where the `add.headers` operation lowercased the header name, causing the configured letter case to be lost when adding new headers to the request.

Core

- **Debugger**: Fixed an issue where Redis command arguments were incorrectly included in Redis spans.
- **Debugger**: Fixed an issue where the time spent during the TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side latency.
- **Debugger**: Fixed an issue where the Konnect Debugger concatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g. `unix:/<prefix>/sockets/am:tls`).
- Fixed an issue where the real client IP from Proxy Protocol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 plain HTTP to HTTPS port) or when the client disconnected prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from `$proxy_protocol_addr` on early request termination, and added `error_page` 497 routing through `kong_error_handler`.

Clustering

- Fixed an issue where Incremental Sync (ICS) would crash and stop further sync attempts.
- Fixed clustering sync reliability issues by immediately failing pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

Dependency

Core

- Bumped lua-resty-healthcheck from 3.1.1 to 3.1.2.

3.13.0.3

R

E

P

st from 0 to 4 for Kafka 3.x. No changes made for Kafka 4.0, as it is already using version 4.

alid legacy ACL group credentials were rejected with 403 when `include_consumer_groups` was enabled

n `guarding_mode` is set to `INPUT` with `allow_masking` enabled.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - By default, this plugin now blocks unrecognized/new filters added to GCP Model Armor.

- ai-mcp-proxy:
    - Fixed an issue where the ACL metric label value was incorrect.
    - Fixed an issue where metrics were not reported to Konnect since 3.13.0.0.
    - Fixed an issue where the latest 2025-11-25 spec defined the Authorization error codes, which are different from the error codes that Kong uses.
- ai-proxy-advanced:
    - Fixed an issue where Azure OpenAI authentication failed during load balancer failover due to token refresh attempts in `balancer_by_lua`. Tokens are now pre-fetched in the `access` phase.
    - Fixed an issue where metrics were missing and v2 API was unavailable in Cohere native format.
    - Fixed an issue where id/created fields were missing in multiple drivers.
    - Fixed an issue where gzip streaming response in OpenAI format was broken.
    - Fixed an issue where the content-type header from non-200 responses to streaming requests was overridden.
    - Fixed an issue where attributes for GenAI spans were incorrect according to https://opentelemetry.io/docs/specs/semconv/gen-ai/gen-ai-spans.
    - Fixed `usage.cost` for OpenAI `llm/v1/files/{file_id}/content` batch output analytics.
- ai-semantic-prompt-guard:
    - Fixed an issue where a vectorDB error occurred during initialization or config reload.
- graphql-rate-limiting-advanced:
    - Fixed an issue where the graphql-rate-limiting-advanced plugin's API threw 500 errors when handling HTTP PUT requests.
- kafka-log:
    - Fixed an issue where cached producers could cause memory leaks.
- oas-validation:
    - Fixed an issue where swagger `formData` parameters were not properly handled.
    - Fixed an issue where `minLength` and `maxLength` constraints defined in a Swagger 2.0 parameter schema were not enforced.
    - Fixed an issue where the validator cache did not account for the `collect_all_errors` setting.
    - Fixed an issue where parameter validation errors caused a runtime crash when `collect_all_errors` was enabled.
- openid-connect:
    - Fixed an issue where the `username` and `password` were used for Redis Sentinel and no authentication was supported for Redis, causing `NOAUTH Authentication Required` errors.
    - Fixed an issue where client-provided headers could be forwarded to upstream services when the corresponding JWT claim was missing from the token. The plugin now clears client headers before processing `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or downstream.
    - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.
- opentelemetry:
    - Fixed an issue where invalid `x-b3-flags` header values caused 500 responses.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after
ts that pass rate limiting, by updating its priority from 915 to 901.

nory LRU cache size.

/etc/kong/kong.logrotate's ownership and permissions; the script now restores ownership to root:root and

h the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` was
dy messages for 501 and 505 status codes.

orrect `latencies.client` values due to not capturing the access end time properly. This fix ensures that
f Active Tracing being active or not.

- Fixed AWS STS throttling during Redis IAM authentication via token caching.
- **vault**: Fixed an issue where `kong.vault.update()` changed the secret to an empty string temporarily after changing a vault config.

Clustering

- Fixed an issue where the DP could spawn multiple sync timers when downgrading sync protocol to v1, which could cause excessive load on the CP and lead to performance degradation.

Kong Manager

- Fixed workspace navigation for workspace names containing non-ASCII characters.

Dependency

Core

- Bumped kong-gql from 0.2.3 to 0.2.4 to fix the null value issue in the GraphQL query syntax that caused a parsed AST error in the degraphql plugin.

Performance

Admin API

- Optimized `GET /kong` endpoint to use `SELECT DISTINCT` for populating `enabled_in_cluster` instead of iterating all plugin rows.

3.13.0.2

Release date 2026/03/10

Bugfix

Clustering

- Fixed an issue where a SQL injection vulnerability existed in the clustering RPC procedure, ensuring safer SQL operations (CVE-2026-1907).

Core

- Apply Nginx patch for a man-in-the-middle upstream response injection (CVE-2026-1642)

Dependency

Core

- Bumped OpenSSL from 3.4.3 to 3.5.5 to address the issue: CVE-2025-15467.

3.13.0.1

Release date 2026/02/01

Bugfix

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ified because of it was disabled in CP.

o run when kong configuration contains vault references.

re cannot fetch a correct credential for connecting to Azure Redis, especially when using Azure Workload

OpenTelemetry traces showed the same hostname for all retry attempts instead of capturing the specific

P

validate message keys when using Confluent Schema Registry.

on failed silently, returning raw string data instead of deserialized objects.

  - Fix issue where Kafka plugin produced records with incorrect `offsetDelta` in the record package.

- **llm**:
  - Fixed an issue where we could not specify the correct dimensions for Gemini embeddings model provider.
- acl:
  - Fixed an issue where the ACL plugin did not check consumer groups from all sources, causing it to incorrectly deny access when consumer groups were set by other plugins like ACE in `ngx.ctx.authenticated_consumer_groups`.
- ai-mcp-oauth2:
  - Fixed an issue where we didn't clear header for absent claim.
- ai-mcp-proxy:
  - Fixed an issue where `tools/list` could expose tools to consumers denied by tool ACL in listener mode.
  - Fixed an issue where PATCHing the tool caused an error.
- ai-proxy-advanced:
  - Fixed an issue where GCP tokens would not be proactively refreshed before expiration.
- aws-lambda:
  - Fixed an issue where Lambda responses with empty or null body caused 500 errors when `is_proxy_integration` is enabled.
- datakit:
  - Fixed xml_to_json node to recognize `text/xml` content-type in addition to `application/xml` for XML request body processing, ensuring RFC 7303 compliance.
- oas-validation:
  - Fixed an issue where response body buffering was not properly disabled when `validate_response_body` was set to `false`.
- opentelemetry:
  - Fixed an issue where the `http.response.status_code` attribute of the `http.server.request.count` metric was incorrectly set to `kong.response.status_code`. Now, the attribute name follows the OTel specification.
  - Fixed an issue where the `kong.upstream.status_code` field of access logs was incorrectly populated with string value instead of integer value when there was one or more retries to the upstream.

3.13.0.0

Release date 2025/12/18

Breaking Change

Admin API

- Fixed an issue where certain record/map fields with an empty-object default value were incorrectly JSON-encoded as arrays.

Feature

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

for the websocket connections of debug sessions.

ations and created a new trace for the log phase spans.

elimiter (available values: `,`, `;`, `|`) to `admin_gui_auth_conf` for OIDC. This option specifies the delimiter
en multiple group values are concatenated in a single claim using a specific delimiter, this configuration
aim value

P request usage.

he anonymous report to count AI usage per model.

kend.

t to GCP CloudSQL Postgres databases. Configurations include `pg_gcp_auth` and

option in `kong.tools.redis` and `kong.enterprise_edition.tools.redis.v2` modules. When this
secure endpoint, the module's `ssl_verify` flag cannot be disabled.

- Added TLS certificate verification enforcement. When tls_certificate_verify option is “on”, certificate verification can’t be disabled by Service or Plugin entities.
- AWS ElastiCache for Redis with AWS IAM Authentication
- Azure Cache for Redis with Azure Microsoft Entra Authentication
- **core**: Added directive ‘proxy_protocol_passthrough’ to allow gateway keep the proxy_protocol header unmodified.
- **Core**: Added SSL verification monitoring when global `tls_certificate_verify` is enabled. You will see a warning if plugins try to disable SSL verification. This will be an error in Kong 3.14.
- **DP Resilience**: added support for “gs://” schema in config sync backup strategy.
- Enabled at-rest keyring encryption for sensitive fields in HashiCorp Vault integration.
- Enabled at-rest keyring encryption for sensitive fields in Event Hooks.
- Google Cloud Memorystore for Redis with GCP IAM Authentication
- **kafka**: Added support for `tls_verify` in the Kafka library.
- **llm**: Added a shared filter to mark requests using AI EE plugins for license enforcement. let ai request not “double-dipping” the API Requests by counting twice.
- **Redis Configuration**: Added vault reference support for Redis `host`, `port`, and `server_name` fields in plugins using Redis configuration (such as `rate-limiting-advanced`).
- Starting from this version, Kong Gateway Enterprise supports using common Cloud Authentication methods to connect to Cloud Redis instances. This includes supporting for the following products:
- When global `tls_certificate_verify` is enabled, service entities cannot disable certificate verification using the `tls_verify` option anymore.

Plugin

- **AI AWS Guardrails Plugin**:
  - Introduced a new mask mode feature in the AWS Guardrails plugin that allows sensitive data to be masked instead of blocked for requests, responses or both.
- **AI Plugins**:
  - Added AI Lakera Guard plugin, to integrate with the safety API provided by https://www.lakera.ai/lakera-guard.
  - Added support for `stream_options` to request server response usage statistics when using SSE streaming response mode.
  - Added support for the Gemini Files operations, through the `llm_format: gemini` native-SDK option.
  - Added xAI provider support to the LLM plugin, enabling integration with xAI’s Grok chat and image generation models.
  - Adds support for AWS Bedrock Bearer Token authentication in Kong AI plugins, allowing secure access to Bedrock models using temporary or long-lived tokens.
  - Added support for video generation with `video/v1/videos/generations` route type for multiple LLM providers.
- **AI**:
  - Added instrumentation for GenAI spans.
- **ais**:
  - Add new partials(vectordb, embeddings and model) in ai plugins configs
- ace:

Consumer Groups

header `X-ACE-Operation-ID`.

ds in ACE plugin.

obal option. When this option is enabled, the plugin’s `storage_config.vault.tls_verify` setting cannot

ds in ACME plugin.

s plugin analytics.

rification when connecting to the bedrock service.

ds in AI AWS Guardrails plugin.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Added support for the global `tls_certificate_verify` option. When enabled globally via `kong.conf`, the plugin's `ssl_verify` config field cannot be set to `false`. This ensures SSL/TLS certificate verification cannot be disabled when global security policy requires it.
  - Enabled at-rest keyring encryption for sensitive fields in AI Azure Content Safety plugin.
- ai-gcp-model-armor:
  - Added block reason and processing latency metrics to GCP Model Armor plugin analytics.
- ai-mcp-proxy:
  - Added support for cookie in OpenAPI spec when doing MCP conversion.
  - Added support for structured output from MCP conversion.
  - Added support for MCP ACL control.
- ai-proxy-advanced:
  - Added load balance, failover and circuit breaker feature for semantic routing.
  - Added missing `least-connections` load balancing algorithm to the AI Proxy Advanced plugin which was missed in the previous release.
  - Added support for Gemini live websocket in the ai-proxy-advanced plugin.
  - Supported health checks and circuit breaker for the load balancer. Added two new fields `max_fails` and `fail_timeout`.
- ai-proxy:
  - Added support for Cerebras - a new AI Provider.
  - Added support for HuggingFace's new serverless API in the AI Proxy plugin, enabling seamless integration and improved compatibility.
  - Added support for native format video generation in Bedrock and Gemini LLM drivers, allowing direct passthrough of provider-specific video generation requests without format conversion.
  - Added support for batch mode of anthropic.
  - Added support for inline batch mode of gemini
  - Added support for batch API of bedrock.
  - Introduced Alibaba Dashscope as new provider.
- ai-rag-injector:
  - Added metadata filtering and ACL (Access Control List) capabilities to the RAG plugin. This feature enables fine-grained control over document retrieval based on metadata attributes and user permissions, supporting both pgvector and Redis vector database backends.
- ai-rate-limiting-advanced:
  - added support to count cost for routes with dynamic AI models.
- ai-request-transformer:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `https_verify` setting cannot be disabled.
- ai-response-transformer:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `https_verify` setting cannot be disabled.
- ai-sanitizer:

to control whether to log the sanitized items, which may contain sensitive data.

gging of AWS Lambda invocation issues.

obal option. When this option is enabled, the plugin's `https_verify` flag cannot be disabled.

toff for failed login attempts.

y to support verifying server certificates. Added support for the `tls_certificate_verify` global option.
ificate_verify setting cannot be disabled.

rifying server certificates. Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- datadog:
  - Added a configuration option to tag the metrics with the Kong route name, or the route ID if the name is empty.
- datakit:
  - Add support for using `application/x-www-form-urlencoded` as the body encoding.
  - Added json_to_xml node to support converting JSON or lua table to XML data.
  - Added support for clearing headers from the service_request and response nodes
  - Added support for dynamic url in `call` node.
  - Added support for using the `call` node in the post-proxy phase.
  - added the `ssl_verify` flag to the call node. This flag allows users to control certificate verification when making HTTPS requests to the configured `url` endpoint. It cannot be disabled when the `tls_certificate_verify` global option is enabled.
  - Added xml_to_json node to support converting XML data to lua table and JSON format.
- forward-proxy:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `https_verify` setting cannot be disabled.
- header-cert-auth:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` flag cannot be disabled.
  - added the flag `ssl_verify` to control certificate verification when connecting to the server of the OCSP responder's URL and to the server of the CRL Distribution Point.
- hmac-auth:
  - Enabled at-rest keyring encryption for sensitive fields in HMAC Auth plugin.
- http-log:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` flag cannot be disabled.
  - added the flag `ssl_verify` to control certificate verification when pushing logs to the configured `http_endpoint`.
- jwt-signer:
  - added support to the `tls_certificate_verify` global option. When this option is enabled the plugin's flags related to certificate verification cannot be disabled.
  - added the flags `access_token_endpoints_ssl_verify` and `channel_token_endpoints_ssl_verify` to switch certificate verification on the related fields.
- kafka-consume:
  - added new config option `ssl_verify` to support verifying server certificates. Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.
- kafka-log:
  - added new config option `ssl_verify` to support verifying server certificates. Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.
- kafka-upstream:
  - added new config option `ssl_verify` to support verifying server certificates. Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

obal option. When this option is enabled and LDAPS is used, the plugin's `verify_ldap_host` setting

obal option. When this option is enabled and LDAPS is used, the plugin's `verify_ldap_host` or

obal option. When this option is enabled, the plugin's `ssl_verify` flag cannot be disabled.

rification when connecting to the server of the OCSP responder's URL and to the server of the CRL

hen `collect_all_errors` is set to `true` in the plugin configuration. Note: Enabling this option with

  - Support readOnly and writeOnly keywords for OpenAPI 3.1.x.

- opa:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.
- openid-connect:
  - added support to the `tls_certificate_verify` global option. When this option is enabled the plugin's flags `ssl_verify`, `tls_client_auth_ssl_verify`, and `session_memcached_ssl_verify` cannot be disabled.
  - added the flags `session_memcached_ssl` and `session_memcached_ssl_verify` to switch certificate verification when connecting to Memcached server.
- opentelemetry:
  - Added support for exporting access logs via OTLP/HTTP protocol to an observability backend (e.g. OpenTelemetry Collector). Please enable this feature by configuring the `access_logs_endpoint` parameter in the OpenTelemetry plugin.
  - Added support for exporting OpenTelemetry metrics via OTLP/HTTP protocol to an observability backend (e.g. OpenTelemetry Collector). Please enable this feature by configuring the `metrics.endpoint` parameter in the OpenTelemetry plugin.
- prometheus:
  - Added a new label `type` for `http_requests_total` and `stream_sessions_total` metrics providing more specific information.
- rate-limiting-advanced:
  - Added `route` support to fields `identifier` and `compound_identifier`.
- request-callout:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting for HTTPS callouts cannot be disabled.
- request-validator:
  - Provided detailed error messages for `oneOf` / `anyOf` subschemas.
- solace-consume:
  - allowed basic auth credentials to be passed from downstream clients.
- solace-log:
  - allowed basic auth credentials to be passed from downstream clients.
- solace-upstream:
  - allowed basic auth credentials to be passed from downstream clients.
- tcp-log:
  - added new config option `ssl_verify` to support verifying server certificates. Added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `ssl_verify` setting cannot be disabled.
- upstream-oauth:
  - added support for the `tls_certificate_verify` global option. When this option is enabled, the plugin's `client.ssl_verify` setting cannot be disabled.
- *datakit*: the `call` node now supports performing requests via a proxy server.

Clustering

ry logs within 100 seconds from last error.

eats on the data plane side.

P

hentication_headers()

K

iders in Redis config.

P

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **AI Plugins**:
  - Fixed an issue where the Gemini provider would not correctly return Model Armor 'Floor' blocking responses to the caller.
  - Fixed an issue where Gemini (Vertex) models didn't return JSON formatted responses when instructed.
  - Fixed an issue where Gemini with native format was not correctly reporting total tokens.
  - Fixed an issue where it was not possible to use "Search Tools" and "extra_body" when calling Gemini models in OpenAI format.
  - Fixed an issue where OpenAI, OpenAI-compatible, and HuggingFace models may have a 0 `completion_tokens` count with SSE streaming responses.
  - Fixed an issue where Bedrock requests with missing model names in the request path would not be properly validated. Now returns a clear error when model name is required in the path but not provided.
  - Fixed an issue where Bedrock 'Nova' models would not correctly run tool calls when combined with thinking text.
  - Fixed an issue where max_request_body_size can't be 0 and set default to 1048576 (1M).
  - Fixed an issue where `log_payloads` configuration would not work when `log_statistics` was disabled in the logging configuration.
- **AI Sanitize**:
  - Fixed an issue where ai-sanitize plugin does not accept non-OpenAI format.
- **GCP Model Armor**:
  - Fixed an issue where short-lived credentials were not being refreshed properly.
  - Fixed an issue where PARTIAL invocationResult was not handled.
- **ai-azure-content-safety, ai-aws-guardrail**:
  - Fixed an issue where the plugins failed to decompress gzip-encoded responses, leading to errors in response handling.
- **ai-proxy, ai-proxy-advanced**:
  - Fix SSE response parsing when the response is gzip-encoded, last chunk was not processed.
- **kafka**:
  - Fixed an issue where consumed messages were not validated against their JSON schema.
  - Fixed an issue where Schema Registry didn't properly validate JSON schemas.
- **kafka-consume, confluent-consume**:
  - Added `enforce_latest_offset_reset` flag to fix incorrect `latest` offset behavior. When `false` (default), maintains backwards compatibility where `latest` acts like `earliest`. When `true`, `latest` correctly starts from end of topic. Also fixed offset commit bug when no records are consumed, preventing feedback loop with true latest behavior.
- **llm**:
  - Fixed an issue where Gemini embeddings driver did not use correct url.
  - Fixed an issue where HuggingFace embeddings driver did not handle response correctly.
  - Fixed an issue where the subrequest implemented in LLM drivers did not handle error properly.
- **response**:
  - Fixed an issue where the `kong.response.exit() didn't interrupt the execution flow of plugins in header_filter phase`. Also added a new configuration `pdk_response_exit_header_filter_early_exit` to control this behavior to avoid silent breaking changes. The current default value is 'off', `it()` is called in header_filter phase.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ould not use the same apikey.

ot be set correctly in some unhappy paths.

not being set properly when authentication failed.

field in the OpenID Connect configuration of ACE auth strategies, which is not supported and could lead

idated for incremental sync in certain scenarios, leading to stale data being served.

handle cases when a referenced key_set does not exist, now returns a clear error message instead of

- ai-aws-guardrails:

  - Fixed an issue where the ai-aws-guardrails metrics could not be recorded.
  - Fixed an issue where the error does not reflect the root cause when the `guardrails_version` did not match the expected pattern.
- ai-gcp-model-armor:
  - Fixed an issue where SDP (Sensitive Data Protection) filter violations were not detected. The plugin now correctly handles both RAI-style results (matchState + confidenceLevel) and SDP-style results (matchState + findings array with infoType and likelihood).
- ai-llm-as-judge:
  - Added validation to prevent disabling `https_verify` when the global `tls_certificate_verify` configuration option is enabled.
- ai-mcp-oauth2:
  - Fixed an issue where the oidc schema was polluted during merging.
  - Fixed an issue where resource without path was not correctly handled.
  - Fixed an issue where there was an unexpected `required: false` in the plugin schema.
  - Fixed an issue where x-forwarded-* headers were not respected.
  - Fixed an issue where MCP-like request was not authenticated.
- ai-mcp-proxy:
  - Fixed an issue where mcp proxy is not parsing default values properly.
  - Fixed an issue when using passthrough-mode, tools without tool ACL do not apply default ACL.
  - Fixed an issue where MCP server could not call upstream when proxy protocol is used.
  - Fixed an issue where failing to fetch tools cache was not handled properly.
  - Fixed an issue where forwarding client headers to upstream could not be disabled.
  - Fixed an issue where trusted ip relative headers were forwarded upstream.
  - Fixed an issue where `tools/call` could accept malformed requests and generate wrong responses.
  - Fixed an issue where path invalid error was not handled properly.
  - Fixed an issue where we could not override the upstream's scheme when converting MCP tool to RESTful API.
  - Fixed an issue where MCP server could not call upstream when self-signed certificate is used in Kong.
  - Fixed an issue where path traversal can be done with path parameter.
- ai-prompt-template:
  - Fixed an issue where non-OpenAI requests were not being processed correctly.
- ai-proxy-advanced:
  - Fixed an issue where the `max_completion_tokens` parameter was not being set correctly for O1 series models (e.g., `o1`, `o3`, `o4`, `gpt-5`).
  - Fixed an issue where the token count for Gemini Vertex embeddings API in OpenAI format was incorrect.
  - Fixed an issue where the native format option did not work correctly for non-openai formats.
  - Fixed an issue where the semantic load balancing with pgvector namespace was not functioning correctly.
  - ground mode.
  - ion is not handled properly for Azure.
  - models via Azure Foundry were not being processed correctly.
  - nd is not properly proxied.
  - model responses were not being processed correctly.
  - ertex embeddings API in native format was incorrect.
  - S entries by preloading DNS for targets.
  - Advanced plugin when using Azure OpenAI
  - ers.
  - n Titan Embed v2 models.
  - Added tool_calls passthrough and preserved finish_reason in Huggingface driver responses.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed an issue where `tls_certificate_verify` configuration was not respected in JSON-RPC and AWS STS calls.
    - Fixed an issue where HuggingFace embedding driver were incorrectly parsed responses from embedding API
    - Fixed an issue where extra inputs were not permitted for huggingface inference provider
    - Fixed an issue where using ai-proxy-advanced in conjunction with a logging plugin (such as file-log) resulting in missing information on the last entry of the balancer "tries" section.
    - Fixed an issue where cohere embedding model on Bedrock returned a bad request error
- ai-rate-limiting-advanced:
    - Fixed an issue where the plugin decreased requests by whole numbers when using Redis. This is an opt-in fix and can be enabled by setting `decrease_by_fractions_in_redis` to true in the plugin configuration.
- ai-request-transformer:
    - Fixed an issue where ai-request-transformer plugin does not accept capture groups for deployment field
- ai-semantic-cache:
    - Fixed an issue where Cohere and Huggingface models did not work with the semantic cache because of the polluted request format.
    - Fixed an issue where the plugin did not report time to first token (TTFT) metrics when hit.
    - Fixed an issue where the plugin did not allow /responses api to be used with Azure.
    - Fixed an issue where cost savings attributes `ai_proxy_cache_cost_savings` were not properly calculated when cache hits.
- ai-semantic-prompt-guard:
    - deprecate config.rules.max_request_body_size with config.max_request_body_size.
- ai-semantic-response-guard:
    - Fixed an issue where a stacktrace error occurred during initialization and teardown.
- confluent-consume:
    - Fixed an issue where the plugin would fail to connect using mTLS.
- confluent:
    - Fixed an issue where `forward_body` dropped request bodies larger than Nginx's default buffer size (16 KB). It now reads up to 1 MB and returns an error if the body can't be fully read.
- datakit:
    - Fixed implicit request node headers field to be correct type.
- file-log:
    - Fixed an issue where the configuration with leading or trailing spaces in the `path` field could cause upgrade failures.
- ip-restriction:
    - Fixed an issue where blocking an IP over TCP would log error: "function cannot be called in preread phase" (#14749)
- kafka-consume:
    - Fixed an issue where the plugin would fail to connect using mTLS.

erminated when there was an error in Schema Registry.

ls to connect certain Kafka clusters using SCRAM-SHA-256 or SCRAM-SHA-512

ificates, causing failures when certificates were updated. Now, the plugin properly reloads updated

may not be properly invalidated, causing stale data to be served.

ache was not isolated by realm.

e ldap server would return 401 instead of 500.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed an issue where path match pattern failed when `()[]` were in the path pattern.

- oas-validation:
    - Fixed an issue where nested references in oneOf / anyOf when they are used with a discriminator are not unfolded.
    - Fixed an issue where YAML 1.1 `null` values were parsed incorrectly.
    - Fixed an issue where parameter data extraction failed when `$` appears in the path pattern.
- openid-connect:
    - Fixed an issue where the issuer mismatch error message for the token's `iss` claim did not reflect the correct token type and expected issuers.
    - Fixed an issue where the `client_credentials`/`authorization_code` auth would not auto-recover if IdP was not accessible during Kong startup.
    - Fixed an issue where TLS client certificate loading failed in non-default workspaces. The certificate lookup now explicitly specifies the plugin's workspace when querying the database during configuration initialization.
    - Fixed an issue where for incremental sync, consumer related caches may not be properly invalidated, causing stale data to be served.
    - Improved claim validation logic to correctly handle timestamp claims (exp, nbf, iat) even when provided as non-numeric types.
- opentelemetry:
    - Fixed an issue where the reference removing did not match the correct property in otel logs.
    - Fixed an issue where the instrumentation started unexpectedly on control planes of hybird mode.
- prometheus:
    - Fixed an issue where the Prometheus plugin would not return DB connections to the connection pool, potentially leading to exhaustion of available connections under high load.
- request-callout:
    - Fix duplicated content-type headers in the callout request when body.custom is enabled and content-type header is also set in headers.
- saml:
    - Fixed an issue that caused a crash when the NameID Format was set to `Unspecified`.
- solace-log:
    - Fixed end-to-end tracing context propagation.
- solace-upstream:
    - Fixed end-to-end tracing context propagation.
- *request-callout*: Ensure that upstream headers are cleared after custom expressions are executed.

Configuration

- Fixed an issue where the restart command read configuration multiple times and failed to stop nginx.
- **balancer**: Fixed an issue where the dns query not stopped when related upstream deleted.
- **template**: Fixed an issue where duplicated "proxy_protocol on" directive in nginx-kong-stream.conf when KONG_NGINX_SPROXY_PROXY_PROTOCOL was set for ssl stream.

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ions to the control plane are not correctly closed in debug sessions.

pans

vents cannot be handled in traditional cluster mode.

eld module.

occur during reconfiguration.

ot reporting headers and payloads when proxy is used to connect to the telemetry endpoint.

increased even for non-ai requests.

be properly populated, causing stale cache to be served.

tatus codes as 000 (or empty) when HTTP/2 clients disconnected during response processing. This w correctly records the actual upstream status code even when the HTTP/2 client disconnects before

- Fixed an issue where Kong Manager failed to generate a cookie when OIDC was enabled due to a missing `password` grant type in `auth_methods`.

- Fixed an issue where migrations could fail due to missing database relations by repositioning basic-auth and key-auth plugin scripts for correct execution order.
- Fixed an issue where `otlp.prepare_logs` did not reset the `spans` field, causing unsupported data exported to OTel backends.
- Fixed an issue where plugin preparation incorrectly retained userdata values; they are now set to `nil` during preparation.
- **Fixed an issue where the partial entities cannot be deleted when cascading delete a workspace.**
- Fixed an issue where unnecessary table allocations were occurring when encoding traces and logs.
- Fixed an issue where upstreams were unhealthy when `healthchecks.threshold` is set to 100.
- Fixed an issue where `Via` header includes dashes `-` in it (like `1.1 kong/3.13.0.0-enterprise-edition`), which is not allowed by RFC 9001 and may cause issues with some .NET HTTP servers. A new `kong.conf` configuration option `via_header_comply_rfc` was added, when enabled, the `Via` header will not include Kong version and dashes (like `1.1 kong`).
- **kong.tools.jose**: Allow arbitrary size RSA keys.

Clustering

- **balancer**: Fixed an issue where the unsupported field `sticky_sessions_cookie_path` was blocking configuration syncing on data planes older than version 3.11.
- **Clustering** Fixed an issue where consumer groups may cause incremental sync to fail at the first sync of a DP.
- Fixed an issue where the data plane did not stop the health checker before cleaning up old configurations, which could lead to resource leaks.
- Fixed an issue where the prometheus metric `kong_control_plane_connected` was missing when the Data Plane side enabled incremental sync.

CLI Command

- Fixed an issue where the CLI command `kong runner` crashes when clustering cert and key are provided as content from environment variables.

PDK

- Fixed an issue where the `kong.log.serialize()` function incorrectly ommitted the `consumer_id` in the `authenticated_entity` log field.
- **log-pdk**: Fixed an issue in log-pdk where log serialization could cause a panic due to incorrect type handling in `set_serialize_value`. Strengthened validation to prevent this.

Dependency

Core

- Bumped lua-resty-healthcheck to version 3.1.1 to remove incorrect deprecation notice on active healthchecks headers.
- Bumped jq from 1.7.1 to 1.8.1.
- Bumped kong-redis-cluster from 1.5.6 to 1.5.7.
- Bumped libexpat from 2.7.1 to 2.7.3.
- Bumped libxml2 from 2.12.10 to 2.15.0.
- Bumped lua-resty-acme from 0.15.0 to 0.16.0.
- Bumped lua-resty-aws from 1.6.0 to 1.7.1.
- Bumped lua-resty-gcp from 0.0.14 to 0.0.16.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

d Redis authentication support.

aintained fork lua-resty-ljsonschema. Bumped version to kong-lua-resty-ljsonschema 1.3.1.

2025-11-04.

Performance

Core

- **core**: Improved performance of deleting expired rows for Postgres versions below 12.
- **core**: Improved performance of Kong core.

Clustering

- Improved the performance of Konnect Analytics on the Data Plane side.

Plugin

- request-transformer:
  - improved performance of the request-transformer plugin.

3.12.0.12

Release date 2026/09/02

Bugfix

Plugin

- ai-mcp-proxy:
  - Fixed an issue where converted HTTPS MCP tool calls lost the original forwarded port on the internal TLS Unix socket.
- ai-proxy-advanced:
  - Fixed a bug where a response from Anthropic could fail to decode when the client's request advertised Brotli support, since Anthropic's edge may then return a Brotli-encoded body that Kong's response decompression, which only understands gzip, can't inflate. Kong now normalizes the outgoing `Accept-Encoding` request header before proxying to any LLM provider: `gzip, identity` when the client accepts gzip, and `identity` otherwise, so a client that cannot inflate gzip is never handed a compressed body on paths where the upstream response passes through unmodified.
- ai-proxy:
  - Fixed streaming token count metrics that reported inflated totals for third-party models hosted on Vertex AI (claude-, mistral-, jamba-*), caused by incremental counts being treated as cumulative.
- jwt:
  - Fixed an issue where the plugin failed to decode standard Base64 secrets that contained `+` or `/` when `config.secret_is_base64` was enabled.
- prometheus:
  - Fixed an issue where `kong_nginx_requests_total` and `kong_nginx_connections_total` duplicated the `http` values under the `stream` subsystem. They are now exported once under `http`.
  - Histogram series are now emitted grouped by label set (all buckets in ascending `le` order, then `_count`, then `_sum`) so the output follows the OpenMetrics interleaving rule. Previously samples were grouped by suffix, which caused strict consumers such as Google Managed Prometheus to drop points.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a permanent failure that started when a broker answered an `ApiVersions` request with an error code, an empty API key set, or a malformed frame. The client cached that result and every later request failed with `ProduceRequest API version not negotiated; broker may be unreachable` until the node restarted, even after the broker recovered. The client now refuses an unusable `ApiVersions` response, keeps the previous negotiation, and retries the negotiation for a topic whose metadata is already cached.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ateway. The Distroless image ships only the Kong binary and its runtime dependencies, with no package
ect to find in a standard Linux distribution.

method and query were reported incorrectly, or could raise an error, when a request hit the error handler

Core

- Fixed an issue where a failed balancer creation leaked the per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.
- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.
- Fixed an issue where `/status/ready` could report a data plane ready before the new router and plugins iterator became active. Kong now reports readiness only after plugin configuration completes and the live objects are installed.
- Fixed an issue where upstream failover could return intermittent 503 responses while a failover target remained healthy.
- On shell-free (distroless) images, the pluginserver query and start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with a clear error on an unterminated quote or `${`, instead of passing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

Clustering

- Fixed an issue where DP syncing failed with large configs. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful full-sync.
- Fixed an issue where heavy packet loss may prevent the full sync from resuming.
- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.

CLI Command

- Hardened stale nginx PID detection to survive PID reuse. On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading `/proc/<pid>/cmdline`, so a reused PID from an ungraceful exit no longer blocks a restart.

Plugin

- jwt-signer:
  - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.
- mocking:
  - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oauth2-introspection:
  - Fixed an authentication bypass where a revoked or expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false` result is now also cached briefly (5 seconds).
- openid-connect:
  - Fixed an issue where the `redirect_uri` parameter was sent to the token endpoint for all grant types. It is now included only for the `authorization_code` grant, as required by RFC 6749.
  - The `ignore_signature` option now logs a warning when it is set, because it turns off JWS signature verification for the listed grants.
- saml:
  - Hardened signature validation against XML Signature Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that names the `ID` of the element it digests and consumes, so a signature can no longer be bound to a different element than the one that provides the identity. A response whose reference URI is empty or not a same-document fragment is now rejected.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a crash that occurred when a plugin set `oauth2` but omitted the `oauth2_client` record. Previously this returned a 500 from the Admin API during e fetching the schema registry OAuth2 token. The absent `oauth2_client` is now handled and falls back to out `oauth2_client` works end-to-end.

K

orts page did not download the report CSV.

D

C

P

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Improved data plane stability when applying a large configuration sync from the control plane, reducing the chance of a worker restart while the update is being processed.
- Optimized performance of clustering payload handling for version compatibility. This reduces CPU usage and transient memory pressure.

Core

- Improved router update performance during incremental sync by updating service data in place instead of triggering a full router rebuild when only service attributes change.
- Reduced config update latency on hybrid data planes by notifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in parallel.

Admin API

- Improved the performance of listing workspaces with `?counter=true`. Entity counts for the whole page are now retrieved with one grouped query per entity type instead of a separate count query per entity type per workspace, while keeping the counts exact.

3.12.0.10

Release date 2026/08/10

Bugfix

Core

- Applied upstream nginx security patches for CVE-2026-42533.
- Applied upstream nginx security patches for CVE-2026-56434.
- Applied upstream nginx security patches for CVE-2026-60005.
- Applied upstream nginx security patches for CVE-2026-42055 and CVE-2026-48142.

Plugin

- saml:
  - Security: Fixed CVE-2026-14917 in the SAML plugin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly, and logging a warning if misconfigured. This prevents a crafted unsigned SAML response from silently bypassing authentication when `validate_assertion_signature` is set to false.

Dependency

Core

- Bumped jq from 1.8.1 to 1.8.2.

3.12.0.9

Release date 2026/07/22

Feature

C

payloads larger than the configured `cluster_max_payload` size to each other, which could avoid the
nprove performance. Emitted a log entry telling the config payload size and the configured

P

S verification. It allows certificate chain and hostname checks against specified CAs instead of global

w legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.

P

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- ai-proxy-advanced:
  - Fixed an issue where unsafe batch id could cause upstream AI provider path traversal.
  - Fixed an issue where an unsafe model name in native format adapters could cause upstream AI provider path traversal.
  - Fixed an issue where unsafe model names could construct upstream AI provider paths.
- ai-request-transformer:
  - Fixed an issue where unsafe model name could cause upstream AI provider path traversal.
- aws-lambda:
  - Fixed an issue where empty JSON array bodies (`[]`) were incorrectly converted to objects (`{}`) when using `is_proxy_integration=true` and `empty_arrays_mode=correct` alongside the exit-transformer plugin.
  - Fixed an issue where a 500 error occurred when `is_proxy_integration` was enabled and the Lambda function returned `null` values in `headers` or `multiValueHeaders`. Null values are now converted to empty strings.
- exit-transformer:
  - Fixed an issue where a response body that was a JSON primitive (e.g. `false`) caused a 500 internal error.
- ip-restriction:
  - Fixed an issue where warn log messages did not include the plugin name, making it difficult to identify the source of blocked requests.
- oas-validation:
  - Fixed an issue where the OAS specification cache failed to distinguish between different values of `api_spec_encoded` for the same specification content.
  - Fixed an issue where path parameters were extracted from the wrong URI segment when `custom_base_path` was used with bare `/{param}` paths.
- opa:
  - Fixed json body processing to recognize `application/json` content-type with charset, ensuring proper handling of JSON request bodies.
- openid-connect:
  - Fixed an inconsistency where `credential_claim` resolved the same claim name to a different value than `upstream_headers_claims` because it consulted the id_token before the access_token, while the upstream headers resolver consulted the access_token first. With IdPs (e.g. Okta) that set distinct `sub` values in the two tokens, `authenticated_entity.id` did not match the upstream header the plugin set on the same request.
  - Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.
  - Fixed an issue where Redis-backed session storage could fail when Redis connections were routed through a proxy.
  - Fixed an issue where consumer groups could not be resolved when the groups claim was only available from the user info endpoint, which resulted in a `500` error.
- session:
  - Fixed an issue where when Postgres strategy is used for session storage, concurrent authentication requests could result in race condition and session creation collisions.
- kafka-upstream, **kafka-log**, **kafka-consume**, **confluent**, **confluent-consume**: Fixed Kafka keepalive pools leaking connections across different SASL credentials and TLS verification settings.
- kafka-upstream, **kafka-log**, **kafka-consume**, **confluent-consume**: Fixed Kafka broker returning failed connections to the keepalive pool, added SSL handshake retry, and improved error propagation.

C

empty stat object instead of null before the first sync has run, keeping the response shape consistent.

plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or eld for those entities, which the declarative config schema rejects as an unknown field.

LS certificate setup failures instead of loading their backup configuration.

able CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable

export config to fallback storage when the control plane uses periodic polling instead of

pending foreign-key references across a failed resume, causing spurious validation failures.

C

d HTTPS traffic.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **Debugger**: Fixed an issue where the default key name `apikey` of the `key-auth` and `key-auth-enc` plugins was not redacted when captured by the Debugger. This has been resolved by adding `apikey` to the list of default redacted headers.
- Fixed an issue where a failure in a `dao:*:post` hook during insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.
- Fixed an issue where cluster mutex did not timeout immediately with `no_wait = true` when node level mutex is being held by another worker.
- Fixed an issue where immediate cosocket SSL handshake errors could be reported as subsequent socket write failures.
- Fixed an issue where Kong Manager users authenticated through IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim changes, or end up with inconsistent mappings during concurrent authentication requests.
- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. Now includes a cert fingerprint for proper connection isolation.
- Fixed an issue where the `KONG_ANALYTICS_FLUSH_INTERVAL` environment variable was not correctly applied to the analytics flush interval.
- **router**: Fixed an issue where WebSocket routes (`ws`/`wss`) returned HTTP 404 when using `router_flavor=expressions` due to incorrect protocol matching expressions being generated.

Configuration

- Increased the default `proxy_ssl_verify_depth` and `lua_ssl_verify_depth` from `1` to `5`.

Dependency

Core

- Bumped kong-lua-resty-ljsonschema from 1.2.0 to 1.3.3.
- Bumped lua-resty-websocket from version 0.4.0.2 to 0.5.0 to solve an issue when handling protocol upgrade failure.
- Bumped OpenSSL from 3.5.6 to 3.5.7.

Performance

Core

- Improved performance of analytics data collection by reducing the default local buffer size.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

3.12.0.8

Release date 2026/06/11

Bugfix

Core

- Applied upstream nginx security patches for limiting the maximum number of headers (CVE-2026-49975).

3.12.0.7

Release date 2026/05/28

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

0701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.

ader handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header `Connection: Content-Length` headers on requests with a body.

: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-

es: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-

3.12.0.6

Release date 2026/05/11

Bugfix

Core

- **Debugger**: Fixed an issue where Redis command arguments were incorrectly included in Redis spans.
- **Debugger**: Fixed an issue where the time spent during the TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side latency.
- **Debugger**: Fixed an issue where the Konnect Debugger concatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g. `unix:/<prefix>/sockets/am:tls`).
- Fixed an issue where the real client IP from Proxy Protocol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 plain HTTP to HTTPS port) or when the client disconnected prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from `$proxy_protocol_addr` on early request termination, and added `error_page` 497 routing through `kong_error_handler`.

Clustering

- Fixed clustering sync reliability issues by immediately failing pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

Plugin

- service-protection:
  - Fixed an issue where the Service Protection plugin could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after Rate Limiting Advanced and only evaluates requests that pass rate limiting, by updating its priority from 915 to 901.

Dependency

Core

- Bumped lua-resty-healthcheck from 3.1.1 to 3.1.2.

3.12.0.5

Release date 2026/04/15

Bugfix

Configuration

- Added `lru_cache_size` to allow configuration of in-memory LRU cache size.

Plugin

- ai-mcp-proxy:
  - Fixed an issue where we didn't keep the port after MCP conversion.
- ai-proxy-advanced:

ng in multiple drivers.

onjunction with a logging plugin (such as file-log) resulting in missing information on the last entry of the

anced plugin's API threw 500 errors when handling HTTP PUT requests.

tokens without dots) caused a 500 Internal Server Error instead of returning 401 Unauthorized.

memory leaks.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where swagger `formData` parameters were not properly handled.
- Fixed an issue where `minLength` and `maxLength` constraints defined in a Swagger 2.0 parameter schema were not enforced.
- Fixed an issue where nested references in `oneOf`/`anyOf` were not unfolded when used with a discriminator.

- openid-connect:
  - Fixed an issue where client-provided headers could be forwarded to upstream services when the corresponding JWT claim was missing from the token. The plugin now clears client headers before processing `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or downstream.
  - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.

Core

- **core**: Resolved an issue where HTTPS was incorrectly detected if multiple `X-Forwarded-Proto` values were provided.
- Fixed an issue where `$remote_addr` was not replaced with the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` was enabled on the proxy listener. Also added proper error body messages for 501 and 505 status codes.
- Fixed an issue where the Active Tracing would report incorrect `latencies.client` values due to not capturing the access end time properly. This fix ensures that the access end timestamp is always recorded regardless of Active Tracing being active or not.
- Fixed an issue where the postinstall script changed the ownership of `/etc/kong/kong.logrotate` to `kong:kong`, causing permission issues when executed by the root user. Ownership now resets to `root:root` after postinstall.
- **vault**: Fixed an issue where `kong.vault.update()` changed the secret to an empty string temporarily after changing vault config.

Clustering

- Fixed an issue where DP may spawn multiple sync timers when downgrading sync protocol to v1, which may cause excessive load on CP and lead to performance degradation.
- Fixed an issue where Incremental Sync (ICS) would crash and stop further sync attempts.

Kong Manager

- Fixed workspace navigation for workspace names containing non-ASCII characters.

Dependency

Core

- Bumped kong-gql from 0.2.3 to 0.2.4 to fix the null value issue in the GraphQL query syntax that caused a parsed AST error in the DeGraphQL plugin.

Performance

Core

- **core**: Improved performance of Kong core by using `ngx.var.http_*` instead of `kong.tools.http.get_header`.

Admin API

- Optimized `GET /kong` endpoint to use `SELECT DISTINCT` for populating `enabled_in_cluster` instead of iterating all plugin rows.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bumped OpenSSL from 3.4.1 to 3.5.5 to address the issue: CVE-2025-15467.

3.12.0.3

Release date 2026/02/26

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where Bedrock requests with missing model names in the request path would not be properly validated. Now returns a clear error when model name is required in the path but not provided.
- **kafka**:
  - Fix issue where Kafka plugin produced records with incorrect `offsetDelta` in the record package.
- **llm**:
  - Fixed an issue where we could not specify the correct dimensions for Gemini embeddings model provider.
- ace:
  - Fixed an issue where users could set the `anonymous` field in the OpenID Connect configuration of ACE auth strategies, which is not supported and could lead to unexpected authentication behavior.
- ai-mcp-oauth2:
  - Fixed an issue where we didn't clear header for absent claim.
- ai-mcp-proxy:
  - Fixed an issue where path traversal can be done with path parameter.
  - Fixed an issue where PATCHing the tool caused an error.
- ai-proxy-advanced:
  - Fixed balancer retry failures caused by expired DNS entries by preloading DNS for targets.
  - Fixed excessive DNS warning logs for template variable endpoints like `$(headers.x_gcp_endpoint)`. These warnings appeared on every configuration change because template variables cannot be resolved until request time.
  - Fixed an issue where metrics were missing and v2 API was unavailable in Cohere native format.
  - Fixed an issue where GCP tokens would not be proactively refreshed before expiration.
- ai-semantic-prompt-guard:
  - Fixed an issue where a vectorDB error occurred during initialization or config reload.
- aws-lambda:
  - added a verbose error level logging for better debugging of AWS Lambda invocation issues.
  - Fixed an issue where Lambda responses with empty or null body caused 500 errors when `is_proxy_integration` is enabled.
- oas-validation:

not properly disabled when `validate_response_body` was set to `false`.

were used for Redis Sentinel and no authentication supported for Redis, causing `NOAUTH Authentication`

lues caused 500 responses.

expectedly on control planes of hybird mode.

tatus codes as 000 (or empty) when HTTP/2 clients disconnected during response processing. This
w correctly records the actual upstream status code even when the HTTP/2 client disconnects before

o run when kong configuration contains vault references.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where the postinstall script changed kong.logrotate permissions; it now resets to 644.

3.12.0.2

Release date 2025/12/10

Feature

Core

- Added a configuration option `authenticated_groups_delimiter` (available values: `,`, `;`, `|`) to `admin_gui_auth_conf` for OIDC. This option specifies the delimiter used to split group values retrieved from JWT claims. When multiple group values are concatenated in a single claim using a specific delimiter, this configuration allows multiple groups to be extracted from that single claim value

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where it was not possible to use “Search Tools” and “extra_body” when calling Gemini models in OpenAI format.
  - Fixed an issue where `log_payloads` configuration would not work when `log_statistics` was disabled in the logging configuration.
- ace:
  - Fixed an issue where the anonymous consumer was not being set properly when authentication failed.
  - Fixed an issue where observability headers could not be set correctly in some unhappy paths.
- ai-mcp-oauth2:
  - Fixed an issue where MCP-like request was not authenticated. Previously, we only authenticated requests that satisfied the MCP spec. As a result, the attacker can bypass this via an MCP-like request. Therefore, now we change to authenticate all the requests.
  - Fixed an issue where the oidc schema was polluted during merging.
  - Fixed an issue where resource without path was not correctly handled.
  - Fixed an issue where x-forwarded-* headers were not respected.
- ai-mcp-proxy:
  - Fixed an issue where failing to fetch tools cache was not handled properly.
  - Fixed an issue where trusted ip relative headers were forwarded upstream.
  - Fixed an issue where `tools/call` could accept malformed requests and generate wrong responses.
  - Fixed an issue where path invalid error was not handled properly.
  - Fixed an issue where we could not override the upstream’s scheme when converting MCP tool to RESTful API.
- ai-proxy-advanced:
  - Fixed intermittent 500 responses from the AI Proxy Advanced plugin when using Azure OpenAI
  - Fixed an issue where Files content analytics extraction is not handled properly for Azure.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Bedrock returned a bad request error

`oxy_cache_cost_savings` were not properly calculated when cache hits.

C

`cky_sessions_cookie_path` was blocking configuration syncing on data planes older than version 3.11.

D

Core

- Bumped lua-resty-healthcheck to version 3.1.1 to remove incorrect deprecation notice on active healthchecks headers.
- Bumped libexpat from 2.7.1 to 2.7.3.

Performance

Core

- **core**: Improved performance of deleting expired rows for Postgres versions below 12.

3.12.0.1

Release date 2025/11/18

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where Bedrock 'Nova' models would not correctly run tool calls when combined with thinking text.
  - Fixed an issue where the Gemini provider would not correctly return Model Armor 'Floor' blocking responses to the caller.
  - Fixed an issue where Gemini (Vertex) models didn't return JSON formatted responses when instructed.
  - Fixed an issue where OpenAI, OpenAI-compatible, and HuggingFace models may have a 0 `completion_tokens` count with SSE streaming responses.
- **GCP Model Armor**:
  - Fixed an issue where PARTIAL invocationResult was not handled.
  - Fixed an issue where short-lived credentials were not being refreshed properly.
- **kafka-consume, confluent-consume**:
  - Added `enforce_latest_offset_reset` flag to fix incorrect `latest` offset behavior. When `false` (default), maintains backwards compatibility where `latest` acts like `earliest`. When `true`, `latest` correctly starts from end of topic. Also fixed offset commit bug when no records are consumed, preventing feedback loop with true latest behavior.
- **llm**:
  - Fixed an issue where Gemini embeddings driver did not use correct url.
  - Fixed an issue where HuggingFace embeddings driver did not handle response correctly.
  - Fixed an issue where the subrequest implemented in LLM drivers did not handle error properly.
- acl:
  - Fixed an issue where the cache was not being invalidated for incremental sync in certain scenarios, leading to stale data being served.
- ai-aws-guardrails:
  - Fixed an issue where the ai-aws-guardrails metrics could not be recorded.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

stream when proxy protocol is used.

pstream could not be disabled.

stream when self-signed certificate is used in Kong.

ertex embeddings API in native format was incorrect.

ertex embeddings API in OpenAI format was incorrect.

ot work correctly for non-openai formats.

ith pgvector namespace was not functioning correctly.

nd is not properly proxied.

n Titan Embed v2 models.

  - Fixed integration with HuggingFace by adding support for the new serverless API, ensuring seamless compatibility.
  - Fixed an issue where HuggingFace embedding driver were incorrectly parsed responses from embedding API
  - Fixed an issue where extra inputs were not permitted for huggingface inference provider
- ai-rate-limiting-advanced:
  - Fixed an issue where the plugin decreased requests by whole numbers when using Redis. This is an opt-in fix and can be enabled by setting `decrease_by_fractions_in_redis` to true in the plugin configuration.
- ai-request-transformer:
  - Fixed an issue where ai-request-transformer plugin does not accept capture groups for deployment field
- ai-semantic-cache:
  - Fixed an issue where Cohere and Huggingface models did not work with the semantic cache because of the polluted request format.
  - Fixed an issue where the plugin did not report time to first token (TTFT) metrics when hit.
- ai-semantic-response-guard:
  - Fixed an issue where a stacktrace error occurred during initialization and teardown.
- confluent-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- datakit:
  - Fixed implicit request node headers field to be correct type.
- file-log:
  - Fixed an issue where the configuration with leading or trailing spaces in the `path` field could cause upgrade failures.
- kafka-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- key-auth-enc:
  - Fixed an issue where for incremental sync, caches may not be properly invalidated, causing stale data to be served.
- oas-validation:
  - Fixed an issue where parameter data extraction failed when `$` appears in the path pattern.
- openid-connect:
  - Fixed an issue where for incremental sync, consumer related caches may not be properly invalidated, causing stale data to be served.
  - Fixed an issue where the `client_credentials`/`authorization_code` auth would not auto-recover if IdP was not accessible during Kong startup.
- request-callout:
  - Fix duplicated content-type headers in the callout request when body.custom is enabled and content-type header is also set in headers.
- saml:
  - Fixed an issue that caused a crash when the NameID Format was set to `Unspecified`.

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

tion multiple times and failed to stop nginx.

l when related upstream deleted.

C

cookie when OIDC was enabled due to a missing `password` grant type in `auth_methods`.

be properly populated, causing stale cache to be served.

eted when cascading delete a workspace.

events cannot be handled in traditional cluster mode.

database relations by repositioning basic-auth and key-auth plugin scripts for correct execution order.

C

- **Clustering** Fixed an issue where consumer groups may cause incremental sync to fail at the first sync of a DP.
- Fixed an issue where the prometheus metric `kong_control_plane_connected` was missing when the Data Plane side enabled incremental sync.
- Fixed an issue where the data plane did not stop the health checker before cleaning up old configurations, which could lead to resource leaks.

Kong Manager

- **Confluent/Kafka plugins**: Fixed an issue where auth mode was not saving correctly in confluent-consume, kafka-consume, kafka-upstream, kafka-log, and confluent plugins.

3.12.0.0

Release date 2025/10/01

Breaking Change

Plugin

- **konnect-application-auth - Konnect user only**:
  - Changed the priority of konnect application auth plugin from 950 to 960. This change is to make sure the execution sequences of the konnect-application-auth plugin and the ACL plugin are as expected. If you're using the konnect-application-auth plugin and other custom plugins that have a dependency with the konnect-application-auth plugin and have a priority between 950 and 960, you may need to adjust the priorities of these custom plugins accordingly or use dynamic plugin ordering.
- kafka-consume:
  - Added `typedefs.no_service` to the `kafka-consume` plugin to prevent binding to a service, as it does not proxy to one. Note that this is a breaking change. If you previously attached a kafka-consume plugin to a service, it will no longer take effect. In such cases, requests will instead be proxied to the upstream configured in the service.

Deprecation

Core

- The lua-resty-kafka module `resty.kafka` inside the Kong Gateway package is now deprecated. Developers can still use it, but it will no longer receive any updates.

Feature

Plugin

- **AI Plugins**:
  - Support relative path for tool paths in MCP plugins.
  - Added a new `endpoint_id` field for GCP Gemini options to support Model Garden.
  - GCP authentication order can now be configured using the `GCP_AUTH_METHOD_ORDER` environment variable. Setting it to `adc` allows the plugin to select Application Default Credentials (ADC) in precedence over Service Account.
- **Kafka**:
  - Supported sending failed messages to dead letter queue topics.
- **ai-mcp**:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ort any API to AI applications via MCP.

same plugin instance.

LLM plugins.

nfluent Schema Registry.

ntent as needed.

  - Add the `ace` plugin.

- ai-aws-guardrails:
  - Add support for AWS IAM role to the ai-aws-guardrails plugin.
- ai-azure-content-safety:
  - Added response guard to filter AI responses for safety and compliance.
- ai-llm-as-judge:
  - Introduced the AI LLM as Judge plugin, enabling you to evaluate an LLM and use the score as a balancer.
- ai-mcp-oauth2:
  - Dropped the `enabled` field, as we already have one in plugin table.
  - Introduced the AI MCP OAuth2 plugin, which protects the MCP traffic with OAuth2.
- ai-mcp-proxy:
  - The plugin now forwards client headers when calling the tool.
- ai-proxy-advanced:
  - Added latency and cost observability to realtime API.
  - Added support for Gemini Rerank in native format.
  - Exposed `input_tokens_details.cached_tokens_details.image_tokens_count` in observability metrics.
- ai-proxy:
  - Added `time_to_first_token` and `request_mode` for observability.
- ai-sanitizer:
  - Added a `block_if_detected` option to reject requests containing PII data instead of sanitizing them. When `block_if_detected` is enabled, requests with detected PII will be blocked with a 400 response containing details about the identified PII types and count.
  - Added support for sanitizing LLM responses.
- ai-semantic-response-guard:
  - Introduced the AI Semantic Response Guard plugin, enabling you to guard LLM responses against semantic content.
- aws-lambda:
  - Added support for AWS API Gateway payload v2. A new configuration field `awsgateway_compatible_payload_version` has been added to control the version of the payload.
- confluent-consume:
  - Added `websocket` mode to the `confluent-consume` plugin.
- datakit:
  - Added caching support via a new `cache` node type.
  - Added support for conditional execution via the `branch` node.
- injection-protection:

path or query.

annel_token_signing to quickly turn on and off token signing or re-signing.

cation: `access_channel_token_(introspection_)optional_claims`. The specified claims are verified

fication: `access_channel_token_(introspection_)required_claims`. The specified claims must be

luding `notbefore`, `issuer`, `subject`, and `audience`.

ing.

/or User-Agent header).

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Added the `peer.service` attribute to the balancer span to link multiple services together. This allows for better tracing of requests that span multiple services.

- rate-limiting-advanced:
  - Added request throttling feature.
- request-callout:
  - The plugin now supports dynamic request URLs in the form of Lua expressions `$(some_lua_expression)`.
- session:
  - Added support for session binding (IP, scheme, and/or User-Agent header).
- solace-log:
  - Added the `solace-log` plugin to support logging to Solace PubSub+ event broker.
- Added vault support to the Datakit plugin. This unblocks common use cases such as storing secrets in Vault and retrieving them as tokens in Datakit.
- solace-consume Added the `solace-consume` plugin, which adds Solace consumption capabilities to Kong.

Clustering

- Added a feature to show the basic incremental sync information on the Data Plane side.

PDK

- Added a new `kong.client.get_aws_vpce_id()` function for returning the VPC ID of the endpoint in the PROXY protocol v2 header PP2_SUBTYPE_AWS_VPCE_ID.
- **PDK**: Added support for calling the `kong.response.get_raw_body()` and `kong.response.set_raw_body()` PDK functions from plugin `:response()` handlers.
- **PDK Request ID**: Added support for retrieving the request ID via the `kong.request.get_id()` function in the PDK.

Core

- Added failover support for upstream targets, allowing load balancing to backup targets when primary targets are offline.
- Added `$kong_upstream_*` Nginx variables to correctly log upstream information in `log_format`.
- Added the new metric `target_id` (Target entity ID) to the Konnect Analytics to improve the observability.
- Added the `/status/timers` endpoint to the Status API for data planes, mirroring the Admin API `/timers` endpoint.
- AI MCP metrics are now reported to Konnect Analytics.
- **Analytics**: Added the `cache.status` field to collect the cache status of proxy-cache plugin.
- Kong now supports Azure authentication methods for connecting to Azure PostgreSQL databases. A new configuration option `pg_azure_auth` as well as corresponding configuration fields `pg_azure_client_id`, `pg_azure_client_secret`, and `pg_azure_tenant_id` have been added to the Kong configuration file. When `pg_azure_auth` is set to `on`, Kong will use Azure authentication methods to connect to Azure PostgreSQL databases.

Default

- **solace**: Added WebSocket transport support for `PERSISTENT` delivery mode.

Kong Manager

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

T Signer plugin.

, parameter for the Key Authentication and Key Authentication Encrypted plugins.

ection plugin.

prise Edition to point to the latest resources.

F

P

rk correctly with Anthropic Claude models.

k for the Bedrock provider when SSE streaming responses.

  - Fixed an issue where Structured Output did not work correctly with Bedrock (Converse API) models.
  - Fixed an issue where `id`, `created`, and/or `finish_reason` fields were missing from or incorrect in Gemini responses.
  - Fixed an issue where Structured-Output requests were ignored.
  - Fixed an issue where Bedrock (Converse API) didn't properly parse multiple `toolUse` tool calls in one turn, on some models.
  - Fixed an issue where the Dimensionality was not sent for Gemini embeddings calls.
- **AI Semantic Plugins**:
  - Fixed an issue by using a separate column to store the namespace instead of including it in the table name to avoid truncation. This change invalidates previous caches created by AI Semantic Cache, AI Semantic Prompt Guard, and AI Proxy Advanced. If a user is using a very long cache TTL, cache warmup is required after this change.
- **Semantic Plugins**:
  - Fixed an issue where Gemini Vertex AI embeddings failed due to incorrect URL construction and response parsing.
- **ai-driver**:
  - Fixed an issue where AI Proxy and AI Proxy Advanced generated incorrect default ports for upstream schemes, leading to 400 errors.
- **ai-proxy, ai-proxy-advanced**:
  - Fix panic in LLM observability when populating token usage for AI Proxy token usage details.
  - Fixed an issue where AI Proxy and AI Proxy Advanced could produce duplicate content in the response when the SSE event was truncated.
  - Fixed an issue where AI Proxy and AI Proxy Advanced could drop content in the response when the SSE event was truncated.
  - Fixed an issue where AI Proxy and AI Proxy Advanced couldn't properly failover to a Bedrock target.
- **ais**:
  - Fixed an issue where the LLM license migration could fail if the license counter contained more than one week of data.
- **embeddings**:
  - Fixed an issue where the `text-embedding-ada-002` model would return a 400 error when the `dimensions` parameter was set.
- **kafka-log, kafka-upstream, kafka-consume**:
  - Added warnings when the strategy is SASL without a mechanism, or the mechanism is set without a SASL strategy.
- ai-aws-guardrails:
  - Fixed the plugin to work with multi-modal content.
- ai-azure-content-safety:
  - Fixed the plugin to work with multi-modal content.
- ai-llm-as-judge:
  - Fix an issue where the ai-llm-as-judge plugin didn't work with non-OpenAI provider configuration.
- ai-prompt-compressor:
  - Fixed the plugin when no model parameter is specified.
- ai-proxy-advanced:

rectly setting the namespace in the `kong.plugin.ctx`, causing the ai-proxy-advanced balancer to retry the

pport Model Garden.

ched incorrectly.

`Unsupported field: seed` when using some inference libraries.

erly validated for certain providers. Note that Bedrock, Gemini Public, and Mistral providers don't support

ls reported malformed requests.

n's lowest latency load balancer could fail to select a target when only one target was available, leading to

ctly parsed.

  - Fixed an issue where the Anthropic provider failed to stream function call responses.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where the Anthropic provider could truncate tokens in streaming responses.
  - Fixed an issue where OpenAI chat completion's `tool_choice` was not converted to Anthropic's.
  - Fixed an issue where the `model` field was incorrect in Gemini responses when a variable was used as the model name.
  - Fixed an issue where the `model` field was missing in OpenAI-format Gemini responses in some situations.
  - Fixed an issue where the Gemini provider did not properly handle multiple tool calls across different turns.
  - Fixed an issue where an empty array was encoded as an empty object in structured output.
  - Fixed an issue where AWS stream parser didn't parse correctly when the frame was incomplete.
  - Fixed an issue where the Anthropic provider mapped the wrong `stop_reason` to Chat and Completions responses.
  - Fixed an issue where AWS Bedrock streaming requests would panic when the ai-semantic-cache plugin was enabled.
  - Fixed an issue where the resource was not being passed correctly when using Azure as the provider.
  - Fixed an issue where AWS Bedrock did not properly handle image editing requests through the `image/v1/images/edits` route.
  - Fixed an issue with Azure Managed Identity credentials causing improper request balancing.
- ai-sanitizer:
  - Fixed an issue where responses could not be redacted correctly when there were multiple redact texts of different lengths sharing the same prefix.
- ai-semantic-cache:
  - Fixed an issue where the DELETE `/ai-semantic-cache` endpoint returned an incorrect 204 status for non-existent keys.
  - Fixed an issue where AI inference requests to non-OpenAI providers in the stream mode were not supported.
  - Fixed an issue where the SSE terminator could have incorrect ending characters.
  - Fixed an issue where the DELETE `/ai-semantic-cache` endpoint returned 500 errors for successful deletions when using pgvector as the vector database.
- ai-semantic-prompt-guard:
  - Fixed the deletion of guard instances and improved error handling.
- ai-semantic-response-guard:
  - Fixed the deletion of guard instances and improved error handling.
- app-dynamics:
  - Fixed an issue where response time measurements to backend services were inaccurate. The plugin now captures the full request duration, including upstream.
- aws-lambda:
  - Fixed an issue where the AWS Gateway compatible payload generated by the plugin did not include the `resource` field in some cases.
- confluent:
  - Fixed an issue where the plugin did not use `conf.cluster_name`.
- datakit:
  - Fixed an issue where an invalid plugin instance could affect other plugins.
  - Fixed plugin config validation to catch more invalid usage of the `property` node.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Status API.

be invalidated correctly.

st bodies larger than Nginx's default buffer size (16 KB). It now reads up to 1 MB and returns an error if

names containing asterisk (*) characters were incorrectly marked as invalid during Kong Manager RBAC
rs (such as `*Dev - EXAMPLE - TOP`) now properly validate and allow role mapping.

locking plugin schema were incorrectly returned as object types instead of array types in response bodies.

- mtls-auth:
  - Fixed an issue where a certificate with an empty DN led to a runtime exception.
- oas-validation:
  - Fixed an issue where certain query strings in URLs that did not conform to the configured OpenAPI specification could cause the plugin to throw an unexpected runtime error and respond with an HTTP 500 error.
- oauth2-introspection:
  - Fixed a high memory usage issue caused by caching negative introspection responses.
  - Fixed an issue where authentication failures did not log messages in the error logs.
- openid-connect:
  - Fixed an issue where `upstream_session_id_header` and `downstream_session_id_header` mistakenly required the `downstream_headers_claims` and `downstream_headers_names` to be present.
- prometheus:
  - Fixed an issue where WebSocket wasn't recorded in the Prometheus metrics.
  - Fixed an issue where Prometheus metrics would show empty consumer identifiers when consumers didn't have a username. The plugin now falls back to using the consumer's `custom_id` when the username is not available, ensuring consumer identification in metrics.
- request-callout:
  - Fixed an issue where expressions could not read headers, query, and URI captures via shortcuts of the form `$(headers.some_header)`.
  - Fixed an issue where `untrusted_lua_sandbox_requires` option was not respected.
  - Fixed regression where the upstream request body would not be forwarded if `upstream.body.custom` was empty.
  - `request.by_lua` code will now run before a request is made or a cached callout is fetched. Previously, it did not run before an attempt to fetch from cache, which did not allow for cache key customization.
  - The plugin now uses additional components to generate the cache key: request consumer, consumer groups, plugin ID, route ID; callout request method, URL, query params, headers, and body.
- response-transformer-advanced:
  - Fixed a failure that occurred when Nginx gzip was enabled for an upstream.
- solace-upstream:
  - Fixed an issue where `forward_body` dropped request bodies larger than Nginx's default buffer size (16 KB). It now reads up to 1 MB and returns an error if the body can't be fully read.
- tls-metadata-headers:
  - Fixed an issue where a certificate with an empty DN led to a runtime exception.
- xml-threat-protection:
  - Fixed an issue where requests without a body were blocked.

Core

- Fixed an issue where stream/proxy listeners could unexpectedly access the router from the control plane. Stream/proxy listeners are now disabled for the control plane.

e metrics were not reported to Konnect.

ss could throw warning messages about the user directive not being used when running as a non-root user.

nts caused cache key problems, preventing Dev Portal from updating the page cache properly in hybrid

DB page by page.

ectly reported as a warning continuously.

tion failures. **vault**: Log more details when vault callback execution fails.

caused DP to fail to start.

ps could crash the data plane during incremental sync.

cessary memory when only incremental data planes were connected.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where the `/clustering/data-planes` endpoint returned an incorrect DP hostname when incremental sync was enabled.
- Fixed an issue where the incremental sync did not work properly for the stream subsystem.
- Fixed redundant configuration exports when multiple data planes were connected concurrently by coalescing triggers and delegating export/broadcast operations to the background push loop.

Admin API

- Fixed an issue where enabling `openid-connect` authentication in Kong Manager would fail due to `authenticated_groups` being a string.
- Fixed an issue where setting credential TTL to null caused an unexpected error.
- Fixed an issue where some schema endpoints in the Admin API didn't include the `ttl` field in GET responses. This includes:
  - `keyauth_enc_credentials`
  - `ratelimiting_metrics`
  - `keyauth_credentials`
  - `oauth2_tokens`
  - `oauth2_authorization_codes`
  - `clustering_data_planes`
  - `acme_storage`
  - `sessions`
  - `audit_requests`
  - `audit_objects`

PDK

- Fixed an issue where PDK returned non-empty header when requests were not proxied to upstream.

Default

- Fixed an issue where the request count was not being updated based on the `license_creation_date` of the license.
- **kafka**: Fixed an issue where the Kafka or Confluent Consume plugins couldn't consume the correct message when the consuming offset of the topic was not 0.
  kafka: Fixed an issue with parsing `ListOffset` Response version 0.

Kong Manager

- Fixed an issue where the vault picker was missing in the Request Callout plugin.

Dependency

CLI Command

- Bumped ca-certificates (debug tool) from 2025-02-25 to 2025-07-15.
- Bumped curl (debug tool) from 8.12.1 to 8.15.0.
- Bumped nghttp2 (debug tool) from 1.65.0 to 1.66.0.

Core

- Bumped kong-redis-cluster from 1.5.5 to 1.5.6. This fixes an issue where Redis connections were not reused properly.
- Bumped libexpat from 2.6.4 to 2.7.1.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

n issue with connecting to an HTTP/1.0 proxy, and to add an enhancement for status code return.

- Bumped LuaRocks from 3.11.1 to 3.12.2.
- Bumped multipart from 0.5.9 to 0.5.10.
- Bumped PCRE2 from 10.45 to 10.46.
- Bumped snappy from 1.2.1 to 1.2.2.
- Bumped xmlua from 1.2.1 to 1.2.2.

Performance

Performance

- **perf**: Improved the efficiency of entity field access on hot path.

Plugin

- prometheus:
  - Optimized fetching of the `kong_db_entities_total` metric by using internal counters instead of the `select count(*)` query through the database. If this metric is found to be of sync with the real row counts, use the command `kong migrations reinitialize-workspace-entity-counters` to re-sync the counts.

Known Issues

Kong Manager

- **Kong Manager**: In the Confluent Consume, Kafka Consume, Kafka Upstream, Kafka Log, and Confluent plugins, setting `schema_registry.confluent.authentication.mode` or `topics.schema_registry.confluent.authentication.mode` to `none` may not be saved after submission.

3.11.0.14

Release date 2026/06/30

Feature

Clustering

- Added a safeguard to prevent the CP or DP from sending payloads larger than the configured `cluster_max_payload` size to each other, which could reduce frequent disconnections and reconnection attempts and improve performance. Emitted a log entry showing the config payload size and the configured `cluster_max_payload` size when this happens.

Bugfix

Plugin

- **kafka-upstream, kafka-log, kafka-consume, confluent, confluent-consume**:
  - Fixed Kafka keepalive pools leaking connections across different SASL credentials and TLS verification settings.
- exit-transformer:
  - Fixed an issue where a response body that was a JSON primitive (e.g. `false`) caused a 500 internal error.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

d from the wrong URI segment when `custom_base_path` was used with bare `/{param}` paths.

ion/json content-type with charset, ensuring proper handling of JSON request bodies.

esolved the same claim name to a different value than `upstream_headers_claims` because it consulted the m headers resolver consulted the access_token first. With IdPs (e.g. Okta) that set distinct `sub` values in the tch the upstream header the plugin set on the same request.

could fail when Redis connections were routed through a proxy.

w legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.

- Fixed an issue that occurs when Postgres strategy is used for session storage, concurrent authentication requests could result in race condition and session creation collisions.

Clustering

- Fixed an issue where a config backup exported by a data plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or `oic_issuers`. The export incorrectly included a `ws_id` field for those entities, which the declarative config schema rejects as an unknown field.
- Fixed an issue where data planes retried unrecoverable TLS certificate setup failures instead of loading their backup configuration.
- Fixed an issue where DP syncing failed with large configs. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful full-sync.
- Fixed an issue where hybrid data planes retried unrecoverable CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable reconnect storms.
- Fixed an issue where incremental DP (sync v2) would not export config to fallback storage when the control plane uses periodic polling instead of notify_new_version.
- Fixed an issue where paginated full sync could keep stale pending foreign-key references across a failed resume, causing spurious validation failures.

Core

- Fixed an issue where a failure in a `dao:*:post` hook during insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.
- Fixed an issue where cluster mutex did not timeout immediately with `no_wait = true` when node level mutex is being held by another worker.
- Fixed an issue where immediate cosocket SSL handshake errors could be reported as subsequent socket write failures.
- Fixed an issue where Kong Manager users authenticated through IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim changes, or end up with inconsistent mappings during concurrent authentication requests.
- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. Now includes a cert fingerprint for proper connection isolation.
- Fixed an issue where the KONG_ANALYTICS_FLUSH_INTERVAL environment variable was not correctly applied to the analytics flush interval.
- Fixed Kafka broker returning failed connections to the keepalive pool, added SSL handshake retry, and improved error propagation.

Dependency

Core

- Bumped kong-lua-resty-ljsonschema from 1.2.0 to 1.3.3.
- Bumped lua-resty-websocket from version 0.4.0.2 to 0.5.0 to solve an issue when handling protocol upgrade failure.
- Bumped OpenSSL from 3.5.6 to 3.5.7.

Performance

Core

- Improved performance of analytics data collection by reducing the default local buffer size.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

maximum number of headers (CVE-2026-49975).

- Applied upstream nginx security patches for CVE-2026-40701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.

- Security: Fixed CVE-2026-6338 in core proxy request header handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header cleanup, preventing CL.0 request smuggling with crafted `Connection: Content-Length` headers on requests with a body.

Dependency

Core

- Bumped libexpat from 2.7.3 to 2.8.1 to address the issues: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-2026-41080, and CVE-2026-45186.
- Bumped OpenSSL from 3.5.5 to 3.5.6 to address the issues: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-2026-28390, and CVE-2026-31789.

3.11.0.11

Release date 2026/05/11

Bugfix

Plugin

- aws-lambda:
  - Fixed an issue where empty JSON array bodies (`[]`) were incorrectly converted to objects (`{}`) when using `is_proxy_integration=true` and `empty_arrays_mode=correct` alongside the exit-transformer plugin.
  - Fixed an issue where a 500 error occurred when `is_proxy_integration` was enabled and the Lambda function returned `null` values in `headers` or `multiValueHeaders`. Null values are now converted to empty strings.

3.11.0.10

Release date 2026/04/27

Bugfix

Core

- **Debugger**: Fixed an issue where Redis command arguments were incorrectly included in Redis spans.
- **Debugger**: Fixed an issue where the time spent during the TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side latency.
- **Debugger**: Fixed an issue where the Konnect Debugger concatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g. `unix:/<prefix>/sockets/am:tls`).
- Fixed an issue where the real client IP from Proxy Protocol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 plain HTTP to HTTPS port) or when the client disconnected prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from `$proxy_protocol_addr` on early request termination, and added `error_page` 497 routing through `kong_error_handler`.

Clustering

- Fixed clustering sync reliability issues by immediately failing pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

S verification. It allows certificate chain and hostname checks against specified CAs instead of global

could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after
ts that pass rate limiting, by updating its priority from 915 to 901.

D

C

3

R

Bugfix

Configuration

- Added `lru_cache_size` to allow configuration of in-memory LRU cache size.

Core

- **core**: Resolved an issue where HTTPS was incorrectly detected if multiple `X-Forwarded-Proto` values were provided.
- Fixed an issue where `$remote_addr` was not replaced with the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` was enabled on the proxy listener. Also added proper error body messages for 501 and 505 status codes.
- Fixed an issue where the Active Tracing would report incorrect `latencies.client` values due to not capturing the access end time properly. This fix ensures that the access end timestamp is always recorded regardless of Active Tracing being active or not.
- Fixed an issue where the postinstall script changed the ownership of `/etc/kong/kong.logrotate` to `kong:kong`, causing permission issues when executed by the root user. Ownership now resets to `root:root` after postinstall.
- **vault**: Fixed an issue where `kong.vault.update()` changed the secret to an empty string temporarily after changing vault config.

Clustering

- Fixed an issue where DP may spawn multiple sync timers when downgrading sync protocol to v1, which may cause excessive load on CP and lead to performance degradation.
- Fixed an issue where Incremental Sync (ICS) would crash and stop further sync attempts.

Plugin

- graphql-rate-limiting-advanced:
  - Fixed an issue where the graphql-rate-limiting-advanced plugin's API threw 500 errors when handling HTTP PUT requests.
- jwt:
  - Fixed an issue where malformed JWT tokens (e.g. tokens without dots) caused a 500 Internal Server Error instead of returning 401 Unauthorized.
- oas-validation:
  - Fixed an issue where `minLength` and `maxLength` constraints defined in a Swagger 2.0 parameter schema were not enforced.
  - Fixed an issue where swagger `formData` parameters were not properly handled.
- openid-connect:
  - Fixed an issue where client-provided headers could be forwarded to upstream services when the corresponding JWT claim was missing from the token. The plugin now clears client headers before processing `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or downstream.
  - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.

Kong Manager

- Fixed workspace navigation for workspace names containing non-ASCII characters.

Dependency

C

issue in the GraphQL query syntax that caused a parsed AST error in the DeGraphQL plugin.

P

C

`var.http_*` instead of `kong.tools.http.get_header`.

A

for populating `enabled_in_cluster` instead of iterating all plugin rows.

3

R

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Bugfix

Clustering

- Fixed an issue where a SQL injection vulnerability existed in the clustering RPC procedure, ensuring safer SQL operations (CVE-2026-1907).

Core

- Apply Nginx patch for a man-in-the-middle upstream response injection (CVE-2026-1642)

Dependency

Core

- Bumped OpenSSL from 3.4.1 to 3.5.5 to address the issue: CVE-2025-15467.

3.11.0.7

Release date 2026/02/26

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where Gemini (Vertex) models didn't return JSON formatted responses when instructed.
  - Fixed an issue where OpenAI, OpenAI-compatible, and HuggingFace models may have a 0 `completion_tokens` count with SSE streaming responses.
  - Fixed an issue where Bedrock requests with missing model names in the request path would not be properly validated. Now returns a clear error when model name is required in the path but not provided.
- **kafka**:
  - Fix issue where Kafka plugin produced records with incorrect `offsetDelta` in the record package.
- **llm**:
  - Fixed an issue where Gemini embeddings driver did not use correct url.
  - Fixed an issue where HuggingFace embeddings driver did not handle response correctly.
- ai-aws-guardrails:
  - Fixed an issue where the ai-aws-guardrails metrics could not be recorded.
- ai-proxy-advanced:
  - Fixed balancer retry failures caused by expired DNS entries by preloading DNS for targets.
  - Fixed an issue where the token count for Gemini Vertex embeddings API in native format was incorrect.
  - Fixed an issue where the token count for Gemini Vertex embeddings API in OpenAI format was incorrect.
  - Fixed excessive DNS warning logs for template variable endpoints like `$(headers.x_gcp_endpoint)`. These warnings appeared on every configuration change because template variables cannot be resolved until request time.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Advanced plugin when using Azure OpenAI

ot work correctly for non-openai formats.

ion is not handled properly for Azure.

nd is not properly proxied.

ers.

parameter was not being set correctly for O1 series models (e.g., `o1`, `o3`, `o4`, `gpt-5`).

n Titan Embed v2 models.

onjunction with a logging plugin (such as file-log) resulting in missing information on the last entry of the

    - Fixed an issue where Cohere and Huggingface models did not work with the semantic cache because of the polluted request format.
- aws-lambda:
    - added a verbose error level logging for better debugging of AWS Lambda invocation issues.
- datakit:
    - Fixed plugin config validation to catch more invalid usage of the `property` node.
- file-log:
    - Fixed an issue where the configuration with leading or trailing spaces in the `path` field could cause upgrade failures.
- kafka-log:
    - Fixed an issue where certificates could not be loaded from non-default workspaces.
- oas-validation:
    - Fixed an issue where response body buffering was not properly disabled when `validate_response_body` was set to `false`.
    - Fixed an issue where nested references in oneOf / anyOf when they are used with a discriminator are not unfolded.
- openid-connect:
    - Fixed an issue where the `username` and `password` were used for Redis Sentinel and no authentication supported for Redis, causing `NOAUTH Authentication Required` errors.
- opentelemetry:
    - Fixed an issue where invalid `x-b3-flags` header values caused 500 responses.
    - Fixed an issue where the instrumentation started unexpectedly on control planes of hybird mode.
- saml:
    - Fixed an issue that caused a crash when the NameID Format was set to `Unspecified`.

Core

- Fixed an issue where Kong Manager failed to generate a cookie when OIDC was enabled due to a missing `password` grant type in `auth_methods`.
- **Clustering** Fixed an issue where consumer groups may cause incremental sync to fail at the first sync of a DP.
- Fixed an issue where Kong incorrectly logged upstream status codes as 000 (or empty) when HTTP/2 clients disconnected during response processing. This occurred when response buffering was enabled. Kong now correctly records the actual upstream status code even when the HTTP/2 client disconnects before receiving the complete response.
- Fixed an issue where `kong health` command would fail to run when kong configuration contains vault references.
- Fixed an issue where migrations could fail due to missing database relations by repositioning basic-auth and key-auth plugin scripts for correct execution order.
- Fixed an issue where the postinstall script changed kong.logrotate permissions; it now resets to 644.

Clustering

- **balancer**: Fixed an issue where the unsupported field `sticky_sessions_cookie_path` was blocking configuration syncing on data planes older than version 3.11.
- Fixed an issue where the prometheus metric `kong_control_plane_connected` was missing when the Data Plane side enabled incremental sync.

D

C

incorrect deprecation notice on active healthchecks headers.

P

C

Postgres versions below 12.

3

R

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Bugfix

Configuration

- **balancer**: Fixed an issue where the dns query not stopped when related upstream deleted.

Core

- Fixed an issue where for incremental sync, consumer group renaming may not be properly populated, causing stale data to be served.
- **event-hooks**: Fixed an issue where the event_hook crud events cannot be handled in traditional cluster mode.

Plugin

- **kafka-consume, confluent-consume**:
  - Added `enforce_latest_offset_reset` flag to fix incorrect `latest` offset behavior. When `false` (default), maintains backwards compatibility where `latest` acts like `earliest`. When `true`, `latest` correctly starts from end of topic. Also fixed offset commit bug when no records are consumed, preventing feedback loop with true latest behavior.
- ai-proxy:
  - Fixed an issue where HuggingFace embedding driver were incorrectly parsed responses from embedding API
- key-auth-enc:
  - Fixed an issue where for incremental sync, caches may not be properly invalidated, causing stale data to be served.
- openid-connect:
  - Fixed an issue where for incremental sync, consumer related caches may not be properly invalidated, causing stale data to be served.
  - Fixed an issue where the `client_credentials`/`authorization_code` auth would not auto-recover if IdP was not accessible during Kong startup.

Clustering

- Fixed an issue where the data plane did not stop the health checker before cleaning up old configurations, which could lead to resource leaks.

Kong Manager

- **Confluent/Kafka plugins**: Fixed issues where Kafka plugins did not work when configured via Kong Manager.

3.11.0.5

Release date 2025/10/23

Bugfix

Plugin

- **llm**:
  - Fixed an issue where the subrequest implemented in LLM drivers did not handle error properly.
- acl:

idated for incremental sync in certain scenarios, leading to stale data being served.

ol_choice was not converted to Anthropic's.

in Gemini responses when a variable was used as the model name.

penAI format response from Gemini provider in some situations.

port for the new serverless API, ensuring seamless compatibility.

d for huggingface inference provider

by whole numbers when using Redis. This is an opt-in fix and can be enabled by setting
lugin configuration.

does not accept capture groups for deployment field

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- ai-sanitizer:
  - Fixed an issue where responses could not be redacted correctly when there were multiple redact texts of different lengths sharing the same prefix.
- ai-semantic-cache:
  - Fixed an issue where the DELETE /ai-semantic-cache returned 500 even deletion was successful, when using pgvector as vectordb.
  - Fixed an issue where the DELETE /ai-semantic-cache with non-existent key returned incorrect status 204.
  - Fixed an issue where AI inference requests to non-OpenAI providers in the stream mode were not supported.
- aws-lambda:
  - Fixed an issue where the AWS Gateway compatible payload generated by the plugin did not include the `resource` field in some cases.
- confluent-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- datakit:
  - Fixed implicit request node headers field to be correct type.
  - Fixed an issue where an invalid plugin instance could affect other plugins.
- jwe-decrypt:
  - Fixed an issue where the key_sets cache could not be invalidated correctly.
- kafka-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- ldap-auth-advanced:
  - Fixed group validation issue where LDAP group names containing asterisk (*) characters were incorrectly marked as invalid during Kong Manager RBAC authentication. Group names with asterisk characters (such as "Dev - EXAMPLE - TOP") now properly validate and allow role mapping.
- oas-validation:
  - Fixed an issue where parameter data extraction failed when `$` appears in the path pattern.
- response-transformer-advanced:
  - Fixed an issue where the failure occurred when Nginx gzip was enabled for upstream.

Configuration

- Fixed an issue where the restart command read configuration multiple times and failed to stop nginx.

Core

- **Fixed an issue where the partial entities cannot be deleted when cascading delete a workspace.**
- **analytics**: multi-model tokens usage metrics were not reported to konnect

Clustering

- Fixed an issue where `/clustering/data-planes` endpoint returns incorrect DP hostname when incremental sync was enabled.

ps could crash the data plane during incremental sync.

A

tation in Kong Manager would fail due to `authenticated_groups` being a string.

K

cted access' banner when trying to create a new role.

3

R

P

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- ai-proxy:
  - Fixed an issue when empty array in structured output was encoded as empty object.

3.11.0.3

Release date 2025/09/05

Feature

Plugin

- **AI Plugins**:
  - GCP authentication order can be configured by `GCP_AUTH_METHOD_ORDER` environment variable. By configuring it to be “adc” allows plugin to select Application Default Credentials (ADC) in precedence over Service Account.
- ai-proxy-advanced:
  - Support for Gemini Rerank in native format

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue Structured Output did not work correctly with Anthropic Claude models.
  - Fixed an issue Structured Output did not work correctly with Bedrock (Converse API) models.
  - Fixed an issue where `id`, `created`, and/or `finish_reason` fields were missing from or incorrect in Gemini responses.
  - Fixed an issue where Structured-Output requests were ignored.
  - Fixed an issue where the Dimensionality was not sent for Gemini embeddings calls.
- **ai-driver**:
  - Fixed an issue where AI Proxy and AI Proxy Advanced generated incorrect default ports for upstream schemes, leading to 400 errors.
- **embeddings**:
  - Fixed an issue where the `text-embedding-ada-002` model would return an 400 error when the `dimensions` parameter was set.
- ai-aws-guardrails:
  - Fixed the plugin to work with multi-modal content.
- ai-azure-content-safety:
  - Fixed the plugin to work with multi-modal content.
- ai-proxy-advanced:
  - Fixed an issue where Mistral models would return `Unsupported field: seed` when using some inference libraries.
  - Skip unknown cached details key from o11y observation.

as valid request for some providers. Bedrock, Gemini Public and Mistral provider don’t accept array input

l reported malformed request.

d the wrong stop_reason to Chat and Completions responses.

tly parsed.

tials causing improper request balancing.

y handle image editing requests through the `image/v1/images/edits` route.

king plugin schema with object type. It should return an array type instead of an object type in the response

- Fixed an issue where authentication failures did not log messages in the error logs.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- prometheus:
  - Fixed an issue where Prometheus metrics would show empty consumer identifiers when consumers don't have a username. The plugin now falls back to using the consumer's custom_id when username is not available, ensuring consumer identification in metrics.
- request-callout:
  - Fix regression where the upstream request body would not be forwarded if `upstream.body.custom` was empty.
- Fix panic in LLM observability when populating token usage for AI Proxy token usage details
- AI Proxy Advanced: Fixed an issue where gemini provider not support model garden

Core

- Fixed an issue where incorrect delimiter in cluster events caused cache key problems, preventing portal from updating page cache properly in hybrid deployment mode.
- Fixed an issue where results were lost when querying LMDB page by page.
- **vault**: Fixed an issue where vault config `base64_decode` caused DP to fail to start.

Clustering

- Fixed an issue where the incremental sync did not work properly for the stream subsystem.

Kong Manager

- Resolved a security issue in the Kong Manager UI.

Dependency

Core

- Bumped lua-muiltipart from 0.5.9 to 0.5.11
- Bumped lua-resty-gcp from 0.0.13 to 0.0.14.

Performance

Plugin

- prometheus:
  - Optimized fetching of the `kong_db_entities_total` metric by using internal counters instead of `select count(*)` query through the database. If this metric is found to be get out of sync with the real row counts, please use the command `kong migrations reinitialize-workspace-entity-counters` to re-sync the counts.

3.11.0.2

Release date 2025/07/28

Bugfix

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

s failed due to incorrect URL construction and response parsing.

ly setting the namespace in the kong.plugin.ctx, causing ai-proxy-advanced balancer retry first target more

cached properly.

e correctly when the frame was incomplete.

ests would panic when the ai-semantic-cache plugin was enabled.

l correctly when using Azure as provider.

  - Fixed an issue where `untrusted_lua_sandbox_requires` option was not respected.
  - `request.by_lua` code will now run before a request is made or a cached callout is fetched. Previously, it did not run before an attempt to fetch from cache, which did not allow for cache key customization.
  - The plugin now uses additional components to generate the cache key: request consumer, consumer groups, plugin ID, route ID; callout request method, URL, query params, headers, and body.
  - Fixed an issue where expressions could not read headers, query, and URI captures via shortcuts of the form `$(headers.some_header)`.
- Fixed an issue where AI Proxy and AI Proxy Advanced can't properly failover to a Bedrock target.
- Fixed an issue where AI Proxy and AI Proxy Advanced might produce duplicate content in the response when the SSE event was truncated.
- Fixed an issue where AI Proxy and AI Proxy Advanced might drop content in the response when the SSE event was truncated.

3.11.0.1

Release date 2025/07/16

Bugfix

Plugin

- **llm**:
  - Fixed an issue where the llm license migration could fail if the license counter contained more than one week of data.
- ai-semantic-cache:
  - Fixed an issue where SSE terminator may not have correct ending characters.
- request-callout:
  - Fixed an issue where `untrusted_lua_sandbox_requires` option was not respected.

Kong Manager

- Fixed an issue where the vault picker was missing in the Request Callout plugin.

Dependency

Core

- Bumped luarocks from 3.11.1 to 3.12.2. This fixes the error `main function has more than 65536 constants`, which was preventing rocks from being published or installed.
- **lua-resty-lmdb**: updated to version 1.6.1.

3.11.0.0

Release date 2025/07/03

Deprecation

Plugin

ute_type. You are encouraged to use new route_types added in version 3.11.x.x and onwards.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ddings' and new route_type 'llm/v1/embeddings'

uests into Solace messages in a Solace queue or topic.

collecting metrics for image and audio generation requests, Responses API, and Realtime API. Added API.

  - add support for the `huggingface` LLM Native format in the ai related plugins.

  - added `model.options.bedrock.performance_config_latency` config parameter, to allow administrators to enforce performanceConfig 'latency' for all AWS Bedrock requests.
  - Added support for the `cohere` LLM format in the ai related plugins. This allows users can directly use the `cohere` rerank API.
  - changed a description for the max_request_body_size schema field to clarify relation to the request body size limit.
  - maps performanceConfig 'latency' field from client requests, for AWS Bedrock requests.
- **llm**:
  - Added support for Bedrock agent SDK, including rerand, converse, converse-stream, retrieveAndGenerate, and retrieveAndGenerate-stream.
- ai-aws-guardrails:
  - Added a new AI plugin that can protect requests and responses to/from AI LLM.
- ai-prompt-compressor:
  - Added a new plugin that compresses prompt information in requests before they are proxied by the AI Proxy or AI Proxy Advanced plugin.
- ai-rate-limiting-advanced:
  - Added support for the `namespace` field in the AI Rate Limiting Advanced plugin. This allows users to specify a ratelimiting namespace for the plugin, Similar to the existing `namespace` field in the `rate-limiting-advanced` plugin.
- confluent:
  - Added support for sending messages with dynamic keys to determine partitions.
- datakit:
  - Added new datakit plugin
- graphql-rate-limiting-advanced:
  - Added support for passing all downstream headers to GraphQL upstream servers in introspection requests.
- kafka-consume:
  - Added WebSocket support to the kafka-consume plugin for consuming Kafka messages.
- kafka-log:
  - Added support for Kafka 4.0. **kafka-upstream**: Added support for Kafka 4.0.
  - Added support for Schema Registry integration with Confluent Schema Registry for AVRO and JSON schemas.
  - Added support for sending messages with dynamic keys to determine partitions.
- kafka-upstream:
  - Added support for sending messages with dynamic keys to determine partitions.
- oas-validation:
  - Added support for validating `multipart/form-data` requests.
- opentelemetry:
  - Added a new configuration field `sampling_strategy` to the plugin. The field allowed you to specify the sampling strategies for OTLP `traces`. Set `parent_drop_probability_fallback` if you want parent-based sampling when the parent span contains a `false` sampled flag, and fallback to probability-based sampling otherwise. Set `parent_probability_fallback` if you want parent-based sampling when the parent span contains a valid sampled flag (`true` ng otherwise. The default value is `parent_drop_probability_fallback` and keeps the instrumentation,

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

rics for ai-rag-injector plugin.

de, and request latency (in milliseconds).

if one isn't provided. The default value is `kong/<kong_version>/request-callout`.

ersions.

rompt field was too short.

Configuration

- Added a configuration option `vault_secrets_lazy_load` to control how vault secrets are loaded. When enabled, plugin options stored as vault secrets are loaded only when needed. This can save memory and reduce startup time but may increase latency the first time a secret is accessed or after it expires. The default value is `off`, preserving the behavior of previous releases.
- Added an optional configuration parameter `admin_gui_hide_konnect_cta` that controls the visibility of the Konnect call-to-action in the Admin GUI.

PDK

- Added a new `kong.request.get_raw_forwarded_path()` function for returning non-normalized `forwarded_path`.
- **jwe**: Supported more encryption algorithms for jwe, including A128KW, A192KW, A256KW, ECDH-ES+A128KW, ECDH-ES+A192KW, ECDH-ES+A256KW, A128GCMKW, A192GCMKW, A256GCMKW.

Core

- Added an optional configuration parameter `service.tls_sans`. This allows to specify additional entries (either DNS names or URIs) for Subject Alternative Names validation of a TLS-based upstream.
- Added certificate-based authentication to the Hashicorp vault backend.
- Added support for OpenTelemetry formatted logs to the (tech preview) Active Tracing feature
- Added support for reporting whether `new_dns_client` and `incremental_sync` were enabled in anonymous reports.
- Added the sticky-sessions load-balancer algorithm.
- **ai**: Change sales counter sync per hour instead of per week.
- **Conjur Vault**: Added support for CyberArk Conjur Vault.
- **Core**: Provided more granular and valuable latency metrics in Kong Gateway, aligning with our Active Tracing measurements.
- Created internal JSON Object Signing and Encryption (JOSE) library to improve performance, features and security of several plugins.
- Logged license model_hours exceeded warning when LLM model usage exceeded the limit.
- Schema map values can now assume null values.
- **vault**: Added support to decode base64 secrets before using them.

Kong Manager

- Added a feature to persist the configuration format preference.
- Added a new UI for the Request Callout plugin.
- Improved the Route configuration form for a better user experience.

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where the Gemini provider could not use Anthropic ‘rawPredict’ endpoint models hosted in Vertex.
  - Fixed unable to use Titan Text Embeddings models.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

did not work for Ollama (llama2) provider.

pecting responses in non-stream mode.

del, temperature, and other user-defined fields.

gs for record of all tried ai targets.

value to hash.

may not be refreshed after the plugin config is updated.

over from other provider to Bedrock.

    - Fixed an issue where latency metric was not implemented for streaming responses.
    - Fixed an issue where SSE terminator may not have correct ending characters.
    - Fixed an issue where the response body for observability may be larger than the real one because of the stale data.
    - Fixed an issue where large request payload was not logged.
    - Fixed an issue where model name did not escape when using AWS Bedrock inference profile.
    - Fixed an issue where AI Proxy returns 403 using gemini provider.
    - Fixed an issue where patterns using multiple capture groups (e.g., `$(group1)/$(group2)`) failed to extract expected matches.
- ai-rate-limiting-advanced:
    - Fixed an issue where AI rate limiting advanced plugin might panic when use redis strategy and sync_rate is set to 0.
- ai-semantic-cache:
    - Fixed an issue where SSE terminator may not have correct ending characters.
- aws-lambda:
    - A warning message was added during schema validation on those plugins that contain a function name that does not comply with AWS Lambda's FunctionName pattern.
- cors:
    - Fixed an issue where requests with empty `origin` headers would cause empty reply from server.
- degraphql:
    - Fixed an issue where the degraphql handle type Boolean was not being assigned correctly in the degraphql router.
- forward-proxy:
    - Marked the `auth_password` in the `forward_proxy` plugin as an `encrypted` field.
- header-cert-auth:
    - Fixed an issue where Header Cert Authentication plugin failed to validate revocation using OCSP when the downstream connection wasn't an SSL connection.
- jwt-signer:
    - Fixed an issue where data planes didn't use the keys passed from the control plane to sign/re-sign.
- jwt:
    - Fixed an issue where the WWW-Authenticate header used an incorrect delimiter, now using a comma as specified by RFC 7235.
- key-auth:
    - Fixed an issue where region was not permitted to be null for remote identity realms.
    - Fixed an issue where external consumers could not be used with traditional Kong deployments, such as with KIC.
- ldap-auth-advanced:
    - Fixed an issue that caused browsers to automatically pop up dialog boxes when authentication failed while `ldap-auth-advanced` was enabled in the Kong Manager.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

king plugin returned an object type instead of an array type in the response body.

n doing the OCSP client cert verification.

responses without a Content-Type header.

e=false couldn't be validated.

nd error" when include_base_path is set to true and the server url is "/".

hird-party IDP to throw an error during logout attempts if the session had already been invalidated. Added
ut request is sent to the correct client, preventing such errors.

edirect_uri errors when `config.redirect_uri` was not configured and the uri path contained spaces.

- prometheus:
  - Fixed an issue where AI latency histogram might miss some calculation.
  - Fixed an issue where the metric `data_plane_config_hash` might not work correctly for incremental sync.
  - Fixed an issue where the control plane failed to expose the `db_entities_total` metric.
- request-callout:
  - Fix issue where a callout response would not be available to response `by_lua` code
  - Fixed an issue where callouts with the same name would be accepted.
  - Fixed an issue where values in `custom` would not accept explicit null values for removal of fields
  - Fixed an issue where callout and upstream request body customizations were not performed when an empty request body was provided; now, an empty JSON body is used and content-type JSON is added to the request.
  - Fixed an issue where callout plugin failed with timeout when `callouts.request.body.custom` is null and `callouts.request.headers.forward` is true.
  - Fixed an issue where the client request body is forwarded to upstream when `upstream.body.forward` is false.
  - Fixed an issue where caching options modified via `by_lua` would apply to all subsequent callouts.
  - Query parameters specified via `callout.request.query` will now correctly replace those in the callout URL.
- request-validator:
  - Fixed an issue where multipart/form-data content type was rejected.
- upstream-timeout:
  - Fixed an issue where `read_timeout`, `write_timeout`, and `connect_timeout` were set to 0, causing a runtime error. After this release, any existing upstream-timeout plugin with these fields set to 0 will automatically update them to the default value of 60000.
- AI Azure Content Safety: Fixed an issue where Blocklist breaches are not correctly parsed.
- Blocked plugins to execute retry logic. Also improve testing funtions
- Fixed an issue where AI Proxy and AI Proxy Advanced would use corrupted plugin config.
- Fixed an issue where an external Python plugin cannot receive the updated value for its referenceable configuration fields.
- Fixed an issue where some of ai metrics was missed in analytics
- Fixed an issue where the rate limiting plugins can’t handle decimal numbers when using Redis strategy.

Core

- Fixed an issue where `ca_certificate` cache was not invalidated when incremental sync was enabled.
- Fixed an issue where upstream connection pooling failed when pool names exceeded 32 characters by applying a patch from upstream OpenResty.
- Fixed several issues that `kong start` and `kong stop` command throw error when the configuration contains vault reference and a prefix directory exists.
- Reverted removal of `translate_backwards` from plugins’ metaschema in order to maintain backward compatibility.
- Added a new configuration, `route_match_calculation`, which included bug fixes for the `traditional_compatible` and the `expressions` router flavors. The default value `original` retained the current `3.x` router matching behavior without changes. The value `strict` enforced the router matching behavior with corrected calculation as follows:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ly downgraded.
ctly captured requests over HTTP routes.
captured requests over TCP or UDP routes.

rategy was not being set correctly in the database.

ence if configurations are set via both deprecated and new names with diverging values

rkspace caused a communication issue between CP and DP on Incremental Config Sync mode.

mes by event_hooks.

nging routes.

plane to stop proxying when incremental config sync is enabled.

before access phase may not take effect

a full sync with incremental enabled would cause inconsistency. It now fails and reinitiating the sync in

dated router data because the `init` phase was skipped.

- Fixed an issue where Pgvector index failed when using special characters in the name.
- Fixed an issue where Plugins were not exported correctly when using Partials in a custom workspace in Hybrid Mode.
- Fixed an issue where the Active Tracing total time was not including connection and handshake times.
- Fixed an issue where the admin API might not fetch the workspace correctly during batch updates.
- Fixed an issue where the delta type was not being validated during incremental sync.
- Fixed an issue where the error logs generated during router rebuilding might be excessively noisy.
- Fixed an issue where the IAM Authentication for RDS failed due to token not being refreshed under certain conditions.
- Fixed an issue where the LMDB database file was created with incorrect permission when running kong cli command with root privileges.
- Fixed an issue where the log lines might be incorrectly logged.
- Fixed an issue where the Lua VM memory size returned by status API is not accurate on the control plane.
- Fixed an issue where the /schemas/plugins/validate endpoint returned a validation error when a plugin was linked to a partial containing a vault key.
- Fixed an issue where the send request time was incorrectly calculated, now it should be more accurate.
- Fixed an issue where users encountered duplicate key violation errors on `rbac_role_entities_pkey` while using the `deck sync` command.
- Fixed an issue where validation incorrectly passed for mis-matched deprecated and new fields if the new field value was false.
- Fixed an issue where validation required all of `timeout` fields (`connect_timeout`, `read_timeout`, `send_timeout`) to have the same value. In reality only `connect_timeout` has to have the same value since that is the value used for generating the `timeout` field in the response if it is missing in the request.
- **kafka**: Fixed an issue where the Kafka SCRAM authentication did not send the server nonce.

Default

- Fixed an issue where adding the `hash_subject` and `store_metadata` fields to the `portal_session_conf` in the Dev Portal was not working as expected.

Clustering

- Fixed an issue where CP may send sync notifications too frequently when incremental sync is enabled.
- Fixed an issue where data plane (DP) might receive incorrect data if the control plane's (CP) configuration version was older than the DP's version.
- Fixed an issue where dataplane nodes sometimes did not recover from sync failures, leading to recursive retries and failures.
- Fixed an issue where debug level logs for incremental sync were insufficient, making debugging more difficult.
- Fixed an issue where events mechanism of clustering might be initialized repeatedly.
- Fixed an issue where LMDB operation failures during incremental sync were not reported.
- Fixed an issue where RPC calls miss the retry mechanism in the clustering system for resilience against transient network issues.
- Fixed an issue where some logs are missing when incremental sync is enabled on the Data Plane side.
- -> Fixed an issue where sometimes the paginated sync incorrectly reports "history lost" and fails.
- Fixed an issue where the Control Plane (CP) would send duplicate sync notifications when configuration changes occurred.
- Fixed an issue where the control plane might not send RPC calls to the data plane correctly in some corner cases.

side might work inefficiently.

onnections incorrectly on the data plane.

workspace during incremental sync may not have been applied correctly.

ssage correctly when incremental sync was enabled.

dly full sync when incremental sync is enabled.

A

delete a non-default role with the same name.

instead of a 401 error when an empty string was provided for the `Kong-Admin-Token` header.

thy before it is actually ready to accept traffic.

t being disallowed for update in the Admin API.

triggers warnings for vitals and portal deprecation.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Performance

- Fixed an issue where rate-limiting library could cause deadlock with Postgres.

CLI Command

- Fixed an issue where the CLI command `kong runner` crashes if using off strategy.

Kong Manager

- Improved the user experience in Kong Manager by fixing various UI-related issues.

Dependency

Core

- Bumped lua-kong-nginx-module from 0.15.1 to 0.15.2.
- Bumped OpenResty from 1.27.1.1 to 1.27.1.2.
- Bumped lua-kong-nginx-module from 0.15.2 to 0.15.3.
- Bumped lua-resty-azure from 1.6.1 to 1.7.0.
- Bumped luarocks from 3.11.1 to 3.12.1. This fixes the error `main function has more than 65536 constants`, which was preventing rocks from being published or installed.

Default

- **Wasm**: removed ngx_wasm_module from default builds.

Performance

Core

- Improved the performance of string splitting functions in Kong's core.
- Improved performance by removing the n+1 query problem for plugins. Plugins are now fetched in bulk with plugins_partials and partials.
- Optimized query of the default workspace by directly accessing LMDB, improving performance.

Plugin

- ai-proxy:
  - Implemented a faster SSE parser.
- jwt:
  - refactored plugin code to be more performant (measured to be at least three times faster).
- oas-validation:
  - Improved performance for OpenAPI 3.1.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

t using strings longer than 256 characters for cache indexing, which could lead to high memory

ce of full syncs.

ged Kong Gateway deployment for more than a week, upgrades from 3.10 versions to 3.11.0.0 will fail due to a license migration issue. This does not affect Konnect deployments. A fix will be provided in 3.11.0.1.

See breaking changes in 3.11 for a temporary workaround.

Core

- The Brotli module is missing from all of the following ARM64 Kong Gateway Docker images:
  - RHEL 9
  - Debian 12
  - Amazon Linux 2
  - Amazon Linux 2023

3.10.0.18

Release date 2026/08/31

Feature

Core

- Added a Distroless Docker image variant for Kong API Gateway. The Distroless image ships only the Kong binary and its runtime dependencies, with no package managers, no shells or any other programs you would expect to find in a standard Linux distribution.

Bugfix

Clustering

- **Clustering**: Improved RPC connection stability during long-running calls: a data plane now cancels a timed-out RPC so a late response cannot disrupt later calls, and the connection is kept alive (within a bound) while calls are still in flight instead of being dropped as a missed heartbeat.
- Fixed an issue where heavy packet loss could prevent a full sync from resuming.
- Fixed an issue where incremental sync delta coalescing merged non-adjacent deltas for the same entity. This could drop an entity's earlier `create` delta and keep only a later update, reordering it after a delta that already referenced it (for example, a plugin instance referencing a cloned plugin by name), causing validation to fail with `plugin '<name>' not enabled`. Coalescing now only merges successive deltas of the same entity, preserving the causal order the data plane relies on.
- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.
- Fixed an issue where the delta coalescing logic on the CP side might work inefficiently.
- Fixed an issue where, with plugin streaming enabled, a single incremental sync validation failure would wipe out all plugin subschemas because the currently loaded plugins were not correctly excluded from the subschema unload.

PDK

- Fixed an issue on HTTP/2 connections where the request method and query were reported incorrectly, or could raise an error, when a request hit the error handler such as a 502 from an upstream connection reset.

Core

- Fixed an issue where a failed balancer creation leaked the per-upstream create lock, so every later creation for the same upstream stalled for 30 seconds.
- Fixed an issue where a healthcheck counter value of 255 crashed health checker creation. Healthcheck counters (`successes`, `tcp_failures`, `timeouts`, `http_failures`) are now limited to 254.

plane as `ready` before the new router and plugins iterator became active. Kong now reports readiness only are installed.

nd start commands now expand `$KONG_PREFIX` / `${KONG_PREFIX}` from the environment and fail fast with ssing the literal `$KONG_PREFIX` to the pluginserver or building a best-effort argv.

C

On Linux, `kong start` now confirms a live PID actually belongs to nginx by reading ful exit no longer blocks a restart.

P

clude the plugin name, making it difficult to identify the source of blocked requests.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed an issue where the plugin failed to set the signed access token header on the upstream request, when `access_token_upstream_header` (or `original_access_token_upstream_header`) had the same name as `channel_token_request_header`.

- jwt:
    - Fixed an issue where the plugin failed to decode standard Base64 secrets that contained `+` or `/` when `config.secret_is_base64` was enabled.
- mocking:
    - Fixed a memory leak where the plugin config cache did not use weak keys due to a misspelled `__mode` metatable field.
- oauth2-introspection:
    - Fixed a high memory usage issue caused by caching negative introspection responses.
    - Fixed an authentication bypass where a revoked or expired token stayed authorized indefinitely instead of being rejected. An authoritative `active:false` result is now also cached briefly (5 seconds).
- openid-connect:
    - Fixed an issue where the `redirect_uri` parameter was sent to the token endpoint for all grant types. It is now included only for the `authorization_code` grant, as required by RFC 6749.
    - The `ignore_signature` option now logs a warning when it is set, because it turns off JWS signature verification for the listed grants.
- prometheus:
    - Fixed an issue where `kong_nginx_requests_total` and `kong_nginx_connections_total` duplicated the `http` values under the `stream` subsystem. They are now exported once under `http`.
    - Histogram series are now emitted grouped by label set (all buckets in ascending `le` order, then `_count`, then `_sum`) so the output follows the OpenMetrics interleaving rule. Previously samples were grouped by suffix, which caused strict consumers such as Google Managed Prometheus to drop points.
- saml:
    - Hardened signature validation against XML Signature Wrapping. The plugin now checks that the signature `Reference URI` is a same-document fragment that names the `ID` of the element it digests and consumes, so a signature can no longer be bound to a different element than the one that provides the identity. A response whose reference URI is empty or not a same-document fragment is now rejected.
- kafka-upstream, **kafka-log**, **confluent**, **kafka-consume**, **confluent-consume**: Fixed a permanent failure that started when a broker answered an `ApiVersions` request with an error code, an empty API key set, or a malformed frame. The client cached that result and every later request failed with `ProduceRequest API version not negotiated; broker may be unreachable` until the node restarted, even after the broker recovered. The client now refuses an unusable `ApiVersions` response, keeps the previous negotiation, and retries the negotiation for a topic whose metadata is already cached.

Kong Manager

- Fixed an issue where the Export button on the Vitals Reports page did not download the report CSV.

Dependency

Core

- Replaced resty-cli with rusty-cli

Performance

Clustering

- Improved data plane stability when applying a large configuration sync from the control plane, reducing the chance of a worker restart while the update is being

A

unter=true. Entity counts for the whole page are now retrieved with one grouped query per entity type
space, while keeping the counts exact.

C

otifying the stream subsystem before applying the update in the HTTP subsystem, so both rebuild in

3

R

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Applied upstream nginx security patches for CVE-2026-42533.
- Applied upstream nginx security patches for CVE-2026-56434.
- Applied upstream nginx security patches for CVE-2026-60005.
- Applied upstream nginx security patches for CVE-2026-42055 and CVE-2026-48142.

Plugin

- saml:
  - Security: Fixed CVE-2026-14917 in the SAML plugin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly (and logging a warning if misconfigured), preventing a crafted unsigned SAML response from silently bypassing authentication under configurations that set `validate_assertion_signature` to false.

Dependency

Core

- Bumped jq from 1.8.1 to 1.8.2.

3.10.0.16

Release date 2026/07/08

Bugfix

Core

- **Debugger**: Fixed an issue where the default key name `apikey` of the `key-auth` and `key-auth-enc` plugins was not redacted when captured by the Debugger. This has been resolved by adding `apikey` to the list of default redacted headers.
- **router**: Fixed an issue where WebSocket routes (`ws`/`wss`) returned HTTP 404 when using `router_flavor=expressions` due to incorrect protocol matching expressions being generated.

Clustering

- Fixed an issue where the control plane could crash while handshaking with a data plane on the legacy clustering endpoint if a non-binary WebSocket frame (such as a ping/pong control frame) arrived before the `basic_info` frame. The control plane now drains and responds to control frames during the handshake until the expected frame is received, instead of dereferencing a nil plugin list.

Plugin

- openid-connect:
  - Fixed an issue where switching clients within a single request (via `client_arg`) failed to use the matching client's JWK.

Performance

Clustering

version compatibility, reducing CPU usage and transient memory pressure.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

plane could not be re-imported if it contained non-workspaceable entities such as `ca_certificates` or
eld for those entities, which the declarative config schema rejects as an unknown field.

LS certificate setup failures instead of loading their backup configuration.

able CP connection failures and ignored Retry-After delays on 429/503 responses, preventing avoidable

pending foreign-key references across a failed resume, causing spurious validation failures.

Core

- Fixed an issue where a failure in a `dao:*:post` hook during insert/update/upsert was silently swallowed, causing the DAO to return `nil` with no error.
- Fixed an issue where immediate cosocket SSL handshake errors could be reported as subsequent socket write failures.
- Fixed an issue where the cluster mutex did not timeout immediately with `no_wait = true` when node level mutex was being held by another worker.
- Fixed an issue where the `KONG_ANALYTICS_FLUSH_INTERVAL` environment variable was not correctly applied to the analytics flush interval.

Plugin

- **kafka-upstream, kafka-log, kafka-consume, confluent, confluent-consume**:
  - Fixed an issue where Kafka keepalive pools would leak connections across different SASL credentials and TLS verification settings.
- openid-connect:
  - Fixed an inconsistency where `credential_claim` resolved the same claim name to a different value than `upstream_headers_claims` because it consulted the `id_token` before the `access_token`, while the upstream headers resolver consulted the `access_token` first. With IdPs (e.g. Okta) that set distinct `sub` values in the two tokens, `authenticated_entity.id` did not match the upstream header the plugin set on the same request.
- Fixed an issue where concurrent authentication requests could cause race conditions and session creation collisions when using the Postgres strategy for session storage.

Configuration

- Increased the default `proxy_ssl_verify_depth` and `lua_ssl_verify_depth` from `1` to `5`.

Dependency

Core

- Bumped kong-redis-cluster from 1.5.7 to 1.6.0.
- Bumped lua-resty-websocket from version 0.4.0.2 to 0.5.0 to solve an issue when handling protocol upgrade failure.
- Bumped OpenSSL from 3.5.6 to 3.5.7.

Performance

Core

- Improved performance of analytics data collection by reducing the default local buffer size.
- Improved router update performance during incremental sync by updating service data in place instead of triggering a full router rebuild when only service attributes change.
- **RBAC**: Improved PostgreSQL performance when deleting Admin API entities by adding an index used to clean up `rbac_role_entities` permissions.

3.10.0.14

Release date 2026/06/11

Bugfix

C

maximum number of headers (CVE-2026-49975).

3

R

F

C

s. Added yielding to prevent CPU stall, and set Konnect mode RPC timeout to 30 seconds for successful

t export config to fallback storage when the control plane used periodic polling instead of

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Fixed an issue where `kong.service.set_tls_cert_key` did not update the upstream keepalive pool name. It now includes a cert fingerprint for proper connection isolation.

Plugin

- Fixed an issue with the Kafka broker returning failed connections to the keepalive pool, added SSL handshake retry, and improved error propagation.

3.10.0.12

Release date 2026/05/28

Bugfix

Core

- Applied upstream nginx security patches for CVE-2026-40701, CVE-2026-40460, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.
- Security: Fixed CVE-2026-6338 in core proxy request header handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header cleanup, preventing CL.0 request smuggling with crafted `Connection: Content-Length` headers on requests with a body.

Dependency

Core

- Bumped libexpat from 2.7.3 to 2.8.1 to address the issues: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-2026-41080, and CVE-2026-45186.
- Bumped OpenSSL from 3.5.5 to 3.5.6 to address the issues: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-2026-28390, and CVE-2026-31789.

3.10.0.11

Release date 2026/05/14

Feature

CLI Command

- Added a `--lts_314_compatibility` option to the `check` command to perform configuration compatibility checks for upgrading to version 3.14.x.x.

Clustering

- Added a safeguard to prevent the CP or DP from sending payloads larger than the configured `cluster_max_payload` size to each other, which could avoid the frequent disconnections and reconnection attempts and improve performance. Emitted a log entry telling the config payload size and the configured `cluster_max_payload` size when this happens.

Bugfix

Core

nts were incorrectly included in Redis spans.

e TLS handshake was incorrectly added to Kong internal latency. This fix moved it to the client side

oncatenated `nil` and errored when accessing Kong Gateway via a Unix socket (e.g.

hrough IDPs such as OIDC or LDAP could retain stale RBAC groups or IDP-sourced roles after claim ncurrent authentication requests.

ol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 ted prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from l added `error_page` 497 routing through `kong_error_handler`.

C

n and stop further sync attempts.

lling pending sync v2 RPC callbacks on disconnect with a `connection closed` error.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- key-auth:
  - Fixed an issue that prevented `region` from being set to `null` for remote identity realms.
- opa:
  - Fixed json body processing to recognize the `application/json` content-type with charset, ensuring proper handling of JSON request bodies.
- openid-connect:
  - Fixed an issue where Redis-backed session storage could fail when Redis connections were routed through a proxy.
- request-validator:
  - Added the `array_length_compat` config option to allow legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.

Dependency

Core

- Bumped kong-lua-resty-ljsonschema from 1.2.0 to 1.3.3.
- Bumped lua-resty-healthcheck from 3.1.1 to 3.1.2.

3.10.0.10

Release date 2026/04/09

Feature

Plugin

- kafka-log and **kafka-upstream**: Added support for Kafka 4.0 to fix the Confluent deprecation.

Bugfix

Core

- **core**: Resolved an issue where HTTPS was incorrectly detected if multiple `X-Forwarded-Proto` values were provided.
- Fixed an issue where in the postinstall script that changed `/etc/kong/kong.logrotate`'s ownership and permissions. The script now restores ownership to `root:root` and sets permissions to 0644.
- Fixed an issue where `$remote_addr` was not replaced with the proxy protocol source address for early nginx error responses (501, 505) when `proxy_protocol` was enabled on the proxy listener. Also added proper error body messages for 501 and 505 status codes.
- Fixed an issue where the Active Tracing would report incorrect `latencies.client` values due to not capturing the access end time properly. This fix ensures that the access end timestamp is always recorded regardless of Active Tracing being active or not.
- **vault**: Fixed an issue where `kong.vault.update()` changed the secret to an empty string temporarily after changing vault config.

Clustering

- Fixed an issue where the DP could spawn multiple sync timers when downgrading sync protocol to v1, which could cause excessive load on the CP and lead to

P

th incorrect `offsetDelta` in the record package.

st from 0 to 4 for Kafka 3.x. No changes made for Kafka 4.0, as it is already using version 4.

anced plugin's API threw 500 errors when handling HTTP PUT request.

d from non-default workspaces.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where `minLength` and `maxLength` constraints defined in a Swagger 2.0 parameter schema were not enforced.

- openid-connect:
  - Fixed an issue where client-provided headers could be forwarded to upstream services when the corresponding JWT claim was missing from the token. The plugin now clears client headers before processing `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or downstream.
  - Fixed DPoP verification for opaque tokens (RFC 9449 Section 6.2) where Kong failed to correctly parse nested `cnf.jkt` claims from introspection responses.
- service-protection:
  - Fixed an issue where the Service Protection plugin could count requests already rejected by Rate Limiting Advanced. Service Protection now executes after Rate Limiting Advanced and only evaluates requests that pass rate limiting, by updating its priority from 915 to 901.

Default

- **kafka-consume** and **confluent-consume**: Fixed an issue where the Kafka Consume and Confluent Consume plugins couldn't consume the correct message when the consuming offset of the topic was not 0. **kafka**: Fixed an issue of parsing ListOffset Response version 0.

Kong Manager

- Fixed workspace navigation for workspace names containing non-ASCII characters.

Dependency

Core

- Bumped kong-gql from 0.2.3 to 0.2.4 to fix the null value issue in the GraphQL query syntax that causes a parsed AST error in the degraphql plugin.

Performance

Core

- **core**: Improved performance of Kong core by using ngx.var.http_* instead of kong.tools.http.get_header.

Admin API

- Optimized `GET /kong` endpoint to use `SELECT DISTINCT` for populating `enabled_in_cluster` instead of iterating all plugin rows.

3.10.0.9

Release date 2026/03/12

Bugfix

Clustering

- Fixed an issue where a SQL injection vulnerability existed in the clustering RPC procedure, ensuring safer SQL operations (CVE-2026-1907).

Core

- Apply Nginx patch for a man-in-the-middle upstream response injection (CVE-2026-1642)

D

C

: CVE-2025-15467.

3

R

P

gging of AWS Lambda invocation issues.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- oas-validation:
  - Fixed an issue where response body buffering was not properly disabled when `validate_response_body` was set to `false`.
- openid-connect:
  - Fixed an issue where the `username` and `password` were used for Redis Sentinel and no authentication supported for Redis, causing `NOAUTH Authentication Required` errors.
- opentelemetry:
  - Fixed an issue where invalid `x-b3-flags` header values caused 500 responses.
  - Fixed an issue where the instrumentation started unexpectedly on control planes of hybird mode.
- request-validator:
  - Added support for specifying JSON schema draft versions.

Core

- Fixed several issues that `kong start` and `kong stop` command throw error when the configuration contains vault reference and a prefix directory exists.
- Fixed an issue where Kong incorrectly logged upstream status codes as 000 (or empty) when HTTP/2 clients disconnected during response processing. This occurred when response buffering was enabled. Kong now correctly records the actual upstream status code even when the HTTP/2 client disconnects before receiving the complete response.
- Fixed an issue where `kong health` command would fail to run when kong configuration contains vault references.

Configuration

- Added `lru_cache_size` to allow configuration of in-memory LRU cache size.

Clustering

- Fixed redundant configuration exports when multiple data planes connect concurrently by coalescing triggers and delegating export/broadcast to the background push loop.
- Fixed an issue where control planes may consume unnecessary memory when only incremental data planes are connected.

Performance

Plugin

- oas-validation:
  - Improved performance for OpenAPI 3.1.

3.10.0.7

Release date 2025/12/09

Bugfix

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

and HuggingFace models may have a 0 `completion_tokens` count with SSE streaming responses.

namespace instead of including it in the table name to avoid truncation. This change invalidates previous ompt Guard and AI Proxy Advanced, if user is using a very long cache TTL, another cache warmup is

nced generated incorrect default ports for upstream schemes, leading to 400 errors.

n LLM drivers did not handle error properly.

idated for incremental sync in certain scenarios, leading to stale data being served.

- ai-proxy:

  - Fixed integration with HuggingFace by adding support for the new serverless API, ensuring seamless compatibility.
  - Fixed an issue where HuggingFace embedding driver were incorrectly parsed responses from embedding API
  - Fixed an issue where extra inputs were not permitted for huggingface inference provider
- ai-rate-limiting-advanced:
  - Fixed an issue where the plugin decreased requests by whole numbers when using Redis. This is an opt-in fix and can be enabled by setting `decrease_by_fractions_in_redis` to true in the plugin configuration.
- ai-request-transformer:
  - Fixed an issue where ai-request-transformer plugin does not accept capture groups for deployment field
- file-log:
  - Fixed an issue where the configuration with leading or trailing spaces in the `path` field could cause upgrade failures.
- key-auth-enc:
  - Fixed an issue where for incremental sync, caches may not be properly invalidated, causing stale data to be served.
- oas-validation:
  - Fixed an issue where nested references in oneOf / anyOf when they are used with a discriminator are not unfolded.
- openid-connect:
  - Fixed an issue where for incremental sync, consumer related caches may not be properly invalidated, causing stale data to be served.
  - Fixed an issue where the `client_credentials`/`authorization_code` auth would not auto-recover if IdP was not accessible during Kong startup.
  - Fixed an issue where the OIDC plugin caused the third-party IDP to throw an error during logout attempts if the session had already been invalidated. Added the `client_id` to the logout URL to ensure the logout request is sent to the correct client, preventing such errors.
- response-transformer-advanced:
  - Fixed an issue where the failure occurred when Nginx gzip was enabled for upstream.

Core

- Fixed an issue where Kong Manager failed to generate a cookie when OIDC was enabled due to a missing `password` grant type in `auth_methods`.
- Fixed an issue where for incremental sync, consumer group renaming may not be properly populated, causing stale data to be served.
- **event-hooks**: Fixed an issue where the event_hook crud events cannot be handled in traditional cluster mode.
- Fixed an issue where migrations could fail due to missing database relations by repositioning basic-auth and key-auth plugin scripts for correct execution order.

Configuration

- **balancer**: Fixed an issue where the dns query not stopped when related upstream deleted.

Clustering

- **Clustering** Fixed an issue where consumer groups may cause incremental sync to fail at the first sync of a DP.
- Fixed an issue where the prometheus metric `kong_control_plane_connected` was missing when the Data Plane side enabled incremental sync.

h checker before cleaning up old configurations, which could lead to resource leaks.

D

C

incorrect deprecation notice on active healthchecks headers.

P

C

Postgres versions below 12.

3

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Release date 2025/10/10

Feature

Core

- Added an optional configuration parameter `service.tls_sans`. This allows to specify additional entries (either DNS names or URIs) for Subject Alternative Names validation of a TLS-based upstream.

Bugfix

Plugin

- **embeddings**:
  - Fixed an issue where the `text-embedding-ada-002` model would return an 400 error when the `dimensions` parameter was set.
- **kafka-consume, confluent-consume**:
  - Added `enforce_latest_offset_reset` flag to fix incorrect `latest` offset behavior. When `false` (default), maintains backwards compatibility where `latest` acts like `earliest`. When `true`, `latest` correctly starts from end of topic. Also fixed offset commit bug when no records are consumed, preventing feedback loop with true latest behavior.
- ai-proxy-advanced:
  - Fixed an issue where AI Proxy Advanced can't failover from other provider to Bedrock.
  - Fixed model name escape or bedrock from request
- ai-proxy:
  - Fixed an issue where OpenAI chat completion's tool_choice was not converted to Anthropic's.
  - Fixed an issue where JSON array data is not correctly parsed.
  - Fixed an issue with Azure Managed Identity credentials causing improper request balancing.
- aws-lambda:
  - Fixed an issue where the AWS Gateway compatible payload generated by the plugin did not include the `resource` field in some cases.
- confluent-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- jwe-decrypt:
  - Fixed an issue where the key_sets cache could not be invalidated correctly.
- kafka-consume:
  - Fixed an issue where the plugin would fail to connect using mTLS.
- ldap-auth-advanced:
  - Fixed group validation issue where LDAP group names containing asterisk (*) characters were incorrectly marked as invalid during Kong Manager RBAC authentication. Group names with asterisk characters (such as "Dev - EXAMPLE - TOP") now properly validate and allow role mapping.
- prometheus:

ow empty consumer identifiers when consumers don't have a username. The plugin now falls back to using
ailable, ensuring consumer identification in metrics.

C

tion multiple times and failed to stop nginx.

C

eted when cascading delete a workspace.

DB page by page.

C

nt returns incorrect DP hostname when incremental sync was enabled.

roperly for the stream subsystem.

- Fixed an issue where adding consumers to consumer groups could crash the data plane during incremental sync.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Admin API

- Fixed an issue where enabling `openid-connect` authentication in Kong Manager would fail due to `authenticated_groups` being a string.

Kong Manager

- Fixed an issue where admins with custom role see 'restricted access' banner when trying to create a new role

3.10.0.5

Release date 2025/08/22

Bugfix

Kong Manager

- Resolved a security issue in the Kong Manager UI.

Performance

Plugin

- prometheus:
  - Optimized fetching of the `kong_db_entities_total` metric by using internal counters instead of `select count(*)` query through the database. If this metric is found to be get out of sync with the real row counts, please use the command `kong migrations reinitialize-workspace-entity-counters` to re-sync the counts.

3.10.0.4

Release date 2025/08/07

Feature

Plugin

- Fixed an issue where the `ai-prompt-decorator` plugin prompt field was too short.

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue where the Gemini provider could not use Anthropic 'rawPredict' endpoint models hosted in Vertex.
- **Semantic Plugins**:
  - Fixed an issue where Gemini Vertex AI embeddings failed due to incorrect URL construction and response parsing.
- ai-proxy-advanced:
  - Fixed an issue where ai-retry phase was not correctly setting the namespace in the kong.plugin.ctx, causing ai-proxy-advanced balancer retry first target more

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

value to hash.

e correctly when the frame was incomplete.

ests would panic when the ai-semantic-cache plugin was enabled.

hen using AWS Bedrock inference profile.

l correctly when using Azure as provider.

gin might panic when use redis strategy and sync_rate is set to 0.

correct ending characters.

  - Fixed an issue where Header Cert Authentication plugin failed to validate revocation using OCSP when the downstream connection wasn't an SSL connection.

- jwt-signer:
  - Fixed an issue where data planes didn't use the keys passed from the control plane to sign/re-sign.
- mocking:
  - Fixed an issue where an empty array set in the mocking plugin schema with object type. It should return an array type instead of an object type in the response body.
- mtls-auth:
  - Fixed an issue where uselss interface is called when doing the OCSP client cert verification.
- oauth2-introspection:
  - Fixed an issue where authentication failures did not log messages in the error logs.
- request-callout:
  - Fixed an issue where `untrusted_lua_sandbox_requires` option was not respected.
  - `request.by_lua` code will now run before a request is made or a cached callout is fetched. Previously, it did not run before an attempt to fetch from cache, which did not allow for cache key customization.
  - The plugin now uses additional components to generate the cache key: request consumer, consumer groups, plugin ID, route ID; callout request method, URL, query params, headers, and body.
  - Fixed an issue where expressions could not read headers, query, and URI captures via shortcuts of the form `$(headers.some_header)`.
  - Fixed an issue where the client request body is forwarded to upstream when `upstream.body.forward` is false.
- Fixed an issue where AI Proxy and AI Proxy Advanced can't properly failover to a Bedrock target.
- AI Azure Content Safety: Fixed an issue where Blocklist breaches are not correctly parsed.
- Fixed an issue where the rate limiting plugins can't handle decimal numbers when using Redis strategy.

Core

- Fixed an issue where the Active Tracing total time was not including connection and handshake times.
- Fixed an issue where the LMDB database file was created with incorrect permission when running kong cli command with root privileges.
- **kafka**: Fixed an issue where the Kafka SCRAM authentication did not send the server nonce.
- Fixed an issue where the /schemas/plugins/validate endpoint returned a validation error when a plugin was linked to a partial containing a vault key.
- Fixed an issue where the Lua VM memory size returned by status API is not accurate on the control plane.

Dependency

Core

- Bumped LuaRocks from 3.11.1 to 3.12.2.
- **lua-resty-lmdb**: updated to version 1.6.1.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- **AI Plugins**:
  - Fixed unable to use Titan Text Embeddings models.
- **ollama**:
  - Fixed an issue where 'tools' and 'RAG' decoration did not work for Ollama (llama2) provider.
- ai-prompt-decorator:
  - Fixed an issue where the plugin would miss the model, temperature, and other user-defined fields.
- ai-proxy-advanced:
  - Fixed an issue where the stale semantic key vector may not be refreshed after the plugin config is updated.
- ai-proxy:
  - Fixed an issue where large request payload was not logged.
  - Fixed an issue where SSE terminator may not have correct ending characters.
  - Fixed an issue where the response body for observability may be larger than the real one because of the stale data.
  - Fixed an issue where patterns using multiple capture groups (e.g., `$(group1)/$(group2)`) failed to extract expected matches.
  - Fixed an issue where AI Proxy returns 403 using gemini provider.
- ai-semantic-cache:
  - Fixed an issue where SSE terminator may not have correct ending characters.
- key-auth:
  - Fixed an issue where external consumers could not be used with traditional Kong deployments, such as with KIC.
- mocking:
  - Fixed an issue where an empty array set in the mocking plugin returned an object type instead of an array type in the response body.
- oas-validation:
  - Fixed an issue where the plugin throws an error for responses without a Content-Type header.
- prometheus:
  - Fixed an issue where AI latency histogram might miss some calculation.
- Fixed an issue where some of ai metrics was missed in analytics

Core

- Added a new configuration, `route_match_calculation`, which included bug fixes for the `traditional_compatible` and the `expressions` router flavors. The default value `original` retained the current `3.x` router matching behavior without changes. The value `strict` enforced the router matching behavior with corrected calculation as follows:
  1. The weight (priority point) of a route was incorrectly downgraded.
  2. A route of HTTPS-only protocol with SNIs incorrectly captured requests over HTTP routes.
  3. A route of TLS-only protocol with SNIs incorrectly captured requests over TCP or UDP routes.

a full sync with incremental enabled would cause inconsistency. It now fails and reinitiating the sync in

ecial characters in the name.

led due to token not being refreshed under certain conditions.

nging routes.

dated router data because the `init` phase was skipped.

y calculated, now it should be more accurate.

lation errors on `rbac_role_entities_pkey` while using the `deck sync` command.

orkspace correctly during batch updates.

ht be initialized repeatedly.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where the incremental sync might not generate correct deltas for non-default workspaces.
- Fixed an issue where the RPC framework might handle connections incorrectly on the data plane.
- -> Fixed an issue where sometimes the paginated sync incorrectly reports “history lost” and fails.
- Fixed an issue where RPC calls miss the retry mechanism in the clustering system for resilience against transient network issues.
- Fixed an issue where dataplane nodes sometimes did not recover from sync failures, leading to recursive retries and failures.
- Fixed an issue where updating an entity in a non-default workspace during incremental sync may not have been applied correctly.
- Fixed an issue where the control plane might not send RPC calls to the data plane correctly in some corner cases.
- Fixed an issue where LMDB operation failures during incremental sync were not reported.

CLI Command

- Fixed an issue where the CLI command `kong runner` crashes if using off strategy.

Dependency

Core

- Bumped luarocks from 3.11.1 to 3.12.1.

Performance

Clustering

- Bumped page size for cluster sync to improve performance of full syncs.
- Improved RPC message queue responsiveness.

3.10.0.2

Release date 2025/05/20

Feature

Configuration

- Added an optional configuration parameter, `admin_gui_hide_konnect_cta`, which controls the visibility of the Konnect call-to-action in Kong Manager.

Core

- Schema map values can now assume null values. This fixes an issue where values in custom schemas wouldn’t accept explicit null values for the removal of fields.

PDK

- Added a new `kong.request.get_raw_forwarded_path()` function for returning the non-normalized `forwarded_path`. This fixes an issue with the OpenID Connect plugin, which was normalizing the path when it shouldn’t.

Bugfix

C

where upstream connection pooling failed when pool names exceeded 32 characters.

ed during incremental sync.

er rebuilds could be excessively noisy.

l.

data plane to stop proxying when incremental config sync was enabled.

P

he deadlocked with Postgres.

tly accessing LMDB, improving performance.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- openid-connect:
  - Fixed an issue which caused an IdP to report invalid `redirect_uri` errors when `config.redirect_uri` was not configured and the URI path contained spaces.
- request-callout:
  - The plugin now logs the request URL, response code, and request latency (in milliseconds).
  - Fixed an issue where a callout response wasn't available to response `by_lua` code.
  - Fixed an issue where caching options modified via `by_lua` would apply to all subsequent callouts.
  - Fixed an issue where callouts with the same name would be accepted.
  - Query parameters specified via `callout.request.query` now correctly replace those in the callout URL.
  - Fixed an issue where values in `custom` wouldn't accept explicit null values for removal of fields.
  - Fixed an issue where callout and upstream request body customizations weren't performed when an empty request body was provided. Now, an empty JSON body is used and `Content-Type: application/json` is added to the request.
  - Fixed an issue where the Request Callout plugin failed with a timeout when `callouts.request.body.custom` was null and `callouts.request.headers.forward` was set to `true`.
- session:
  - Fixed an issue where boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) stored the session's metadata in the database. This also resolves an issue with Dev Portal, where adding these fields to `portal_session_conf` wasn't working as expected.

Admin API

- Fixed an issue where the data plane (DP) could report a healthy status before it was actually ready to accept traffic.

Clustering

- Fixed an issue where debug level logs for incremental sync were insufficient, making debugging more difficult.
- Fixed an issue where some logs were missing when incremental sync was enabled on the data plane side.
- Fixed an issue where the data plane could repeatedly trigger a full sync when incremental sync was enabled.

3.10.0.1

Release date 2025/04/15

Feature

Kong Manager

- Added a new UI for the Request Callout plugin.

Bugfix

Core

- Fixed an issue where `ca_certificate` cache was not invalidated when incremental sync was enabled.

rategy was not being set correctly in the database.

ence if configurations are set via both deprecated and new names with diverging values

ields (`connect_timeout`, `read_timeout`, `send_timeout`) to have the same value. In reality only
s the value used for generating the `timeout` field in the response if it is missing in the request.

P

ig_hash might not work correctly for incremental sync.

ould use corrupted plugin config.

C

rect data if the control plane's (CP) configuration version was older than the DP's version.

- Fixed an issue where CP may send sync notifications too frequently when incremental sync is enabled.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where the Control Plane (CP) would send duplicate sync notifications when configuration changes occurred.
- Fixed an issue where validation might not report error message correctly when incremental sync was enabled.

Known Issues

Core

- A bug originating from the underlying OpenResty platform affects connection pooling in certain circumstances. This issue impacts transaction handling and negatively impacts performance, resulting in reduced request rates and increased latencies. Versions 3.10.0.0 and 3.10.0.1 are both affected. To resolve this regression, please upgrade to Kong Gateway 3.10.0.2 or later.

3.10.0.0

Release date 2025/03/27

Breaking Change

Plugin

- **AI Plugins**:
  - Changed the serialized log key of AI metrics from `ai.ai-proxy` to `ai.proxy` to avoid conflicts with metrics generated from plugins other than AI Proxy and AI Proxy Advanced. If you are using logging plugins (for example, File Log, HTTP Log, etc.), you will have to update metrics pipeline configurations to reflect this change.
- **kong.service.request.clear_query_arg**:
  - Changed the encoding of spaces in query arguments from + to `%20` as a short-term solution to an issue that some users are reporting. While the + character is the correct encoding of space in querystrings, Kong uses `%20` in many other APIs (inherited from Nginx / OpenResty).
- ai-rate-limiting-advanced:
  - `window_size` and `limit` now require an array of numbers instead of a single number. If you configured the plugin before 3.10 and use `kong migrations` to upgrade to 3.10, it will be automatically migrated to use the array.
- openid-connect:
  - Fixed an issue where forbidden requests were redirected to `unauthorized_redirect_uri` if configured. After the fix, forbidden requests will be redirected to `forbidden_redirect_uri` if configured.

Core

- Free mode is no longer available. Starting Kong without a license will now function the same as Kong with an expired license.

Deprecation

Plugin

- **AI Plugins**:
  - Deprecated `preserve` mode in `config.route_type`. Use `config.llm_format` instead. The `preserve` mode setting will be removed in a future release.

Feature

C

eader to Gateway, which controls the Content-Security-Policy (CSP) header served with Admin GUI
-in by setting it to `"on"`.

a X.509 Certificate Thumbprint to identify the key.

m the OpenResty upstream openresty/lua-resty-core#460.

oxy pass balancer is recreated.

als enable users to define shared configuration for Redis.

ng.

l plugins.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- **AI Plugins**:
  - Added the `huggingface`, `azure`, `vertex`, and `bedrock` providers to embeddings. They can be used by the ai-proxy-advanced, ai-semantic-cache, ai-semantic-prompt-guard, and ai-rag-injector plugins.
  - Allow authentication to Bedrock services with assume roles in AWS.
- **ai**:
  - Added support for boto3 SDKs for Bedrock provider, and for Google GenAI SDKs for Gemini provider.
- ai-proxy-advanced:
  - Added cost to `tokens_count_strategy` when using the lowest-usage load balancing strategy.
  - Added new `priority` balancer algorithm, which allows setting apriority group for each upstream model.
  - Added the ability to set a catch-all target in semantic routing.
  - Added the `failover_criteria` configuration option, which allows retrying requests to the next upstream server in case of failure.
- ai-rag-injector:
  - Added a new plugin which allows automatically injecting documents to simplify building RAG pipelines.
- ai-rate-limiting-advanced:
  - Added support for allowing multiple rate limits for the same providers.
- ai-sanitizer:
  - Added a new plugin that can sanitize the PII information in requests before the requests are proxied by the AI Proxy or AI Proxy Advanced plugin.
- confluent:
  - Added support for message manipulation with the new configuration field `message_by_lua_functions`.
  - Added support for sending messages to multiple topics with `topics_query_arg`, and enabled topic allowlisting with `allowed_topics`.
- cors:
  - Added an option to skip returning the Access-Control-Allow-Origin response header when requests don't have the Origin header.
- json-threat-protection:
  - Added the schema field `allow_duplicate_object_entry_name` to allow or disallow duplicate object keys in JSON payloads. When set to `false`, the plugin will reject JSON payloads with duplicate object keys. The default value is `true`, which is same as the previous behavior.
- kafka-upstream:
  - Added support for message manipulation with the new configuration field `message_by_lua_functions`.
  - Added support for sending messages to multiple topics with `topics_query_arg`, and enabled topic allowlisting with `allowed_topics`.
- oas-validation:
  - Added support for `oneOf`, `anyOf`, `allOf`, and `not` keywords.
  - Added support for the `discriminator` keyword in OpenAPI specs.
- opentelemetry:
  - This plugin now supports instana headers in propagation.

es.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ntrol plane.

of Proxy-Wasm metrics.

es complex request augmentation and internal authentication.

ect (default `false`) and `store_metadata` (default `false`) to store the session's metadata in the database.

adds Kafka consumption capabilities to Kong.

- dynamic control upstream tls when kong.service.request.set_scheme was called

- **jwe**: JWE now supports the following encryption algorithms: A128GCM, A192GCM, A128CBC-HS256, A192CBC-HS384, A256CBC-HS512.

Clustering

- Added a feature to store the last sync time on the Data Plane side.
- Added support for incremental config sync for hybrid mode deployments. Instead of sending the entire entity config to data planes on each config update, incremental config sync lets you send only the changed configuration to data planes.

Default

- Added a new feature to invalidate the admin's or the developer's related session while changing the password.

Admin API

- Updated `/license/report` endpoint to include counts for Kafka consumption, Confluent Kafka consumption, and Confluent production.

Kong Manager

- Add Redis shared configuration support in Plugins.
- Kong Manager now returns to the previous page upon canceling plugin editing.
- Kong Manager now shows the scope option in gray when it can't be changed.

Bugfix

Core

- Added a patch for `kong.resty.set_next_upstream()` to control the next upstream retry logic on the Lua side.
- Fixed an issue where a GET request to the Admin API root `/` path would return a 500 server error.
- Fixed an issue where a valid declarative config with certificate or SNI entities couldn't be loaded in DB-less mode.
- Fixed an issue where consistent hashing did not correctly handle hyphenated-Pascal-case headers, leading to uneven distribution of requests across upstream targets.
- Fixed an issue where `POST /config?flatten_errors=1` could return a JSON object instead of an empty array.
- Fixed an issue where `POST /config?flatten_errors=1` could not return a proper response if the input contained duplicate consumer credentials.
- Fixed an issue where `socket_path` permissions were not correctly set to `755` when the umask setting did not give enough permissions.
- Fixed an issue where targets couldn't be removed from the DNS query if they were deleted or updated via the Admin API.
- Fixed an issue where the error reason wasn't thrown when parsing the certificate from vault.
- Fixed an issue where the new DNS client did not correctly handle the timeout option in resolv.conf.
- Fixed an issue where the schema library would error with a nil reference if an entity checker referred to a nonexistent field.
- Fixed an issue where the `tls_verify`, `tls_verify_depth`, and `ca_certificates` properties of a service were not included in the upstream keepalive pool name.
- Fixed potential connection leaks when the data plane failed to connect to the control plane.
- Added an optional configuration parameter `admin_gui_csp_header_value` to Gateway, which controls the value of the Content-Security-Policy (CSP) header served with Admin GUI (Kong Manager).

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

e in the AI driver.

the value extracted during tracing headers propagation.

vault reference occasionally didn't get refreshed on time when initialized with an invalid string.

a DELETE request with `Content-Type: application/json` and no body was made.

uests and ETag in responses would result in a bad case in the response phase.

se RDS IAM authentication when an old worker was decommissioned.

orrectly cached indefinitely in the new DNS client.

nts, caused by the normalized table not including values of type number or boolean.

when failing to connect to an upstream by websocket.

Kong AI Gateway.

re access phase may not take effect

- Fixed an issue where the PEM-formatted private keys in the keys entity were not encrypted when keyring was enabled.
- **Vault**: Updated the AWS Vault supported regions list to the latest.

Plugin

- **AI Plugins**:
  - Fixed an issue where AI upstream URL trailing would be empty.
  -
- **authentications**:
  - Improved the error message which occurred when an anonymous consumer was configured but did not exist.
- ai-proxy-advanced:
  - Fixed an issue where the ai-proxy-advanced plugin identity running failed in retry scenarios.
  - Fixed an issue where the ai-proxy-advanced plugin failed to failover between providers of different formats.
- ai-proxy:
  - Fixed a bug in the Azure provider where `model.options.upstream_path` overrides would always return a 404 response.
  - Fixed a bug where response streaming in Gemini and Bedrock providers was returning whole chat responses in one chunk.
  - Fixed a bug where Azure streaming responses would be missing individual tokens.
  - Fixed a bug with the Gemini provider, where multimodal requests (in OpenAI format) would not transform properly.
  - Fixed an incorrect error thrown when trying to log streaming responses.
  - Fixed an issue where Gemini streaming responses were getting truncated and/or missing tokens.
  - Fixed preserve mode.
  - Fixed tool calls not working in streaming mode for Bedrock and Gemini providers.
- ai-semantic-cache:
  - Fixed an issue where the Refresh header wasn't properly sent to the client.
  - Fixed issue where the SSE body may have extra trailing.
- ai-semantic-prompt-guard:
  - Fixed an issue where Kong Gateway was not able to reconfigure the plugin when using DB-less mode.
- app-dynamics:
  - Fixed a segmentation fault caused by a missing destructor call on process exit.
- aws-lambda:
  - Fixed an issue that occurred when `is_proxy_integration` was enabled, where Kong's response could behave incorrectly when the response was changed after the execution of the AWS Lambda plugin. The Content-Length header in the lambda function response is now ignored by the AWS Lambda plugin.
- file-log:
  - Fixed an issue where an error would occur when there were spaces at the beginning or end of a path.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

as empty in logs when using the `forward-proxy` plugin.

proxy-cache-advanced.

on-`POST`/`PUT`/`PATCH` requests.

upsert jwks if the jwks contains extra custom fields.

the first null character.

    - Fixed an issue where random delays were out of range.

- oas-validation:
    - Fixed an issue where query params without values caused an assertion failure.
- pre-function:
    - Fixed an issue where a duplicate `protocols` field was accidentally added to the `pre-function` schema.
- rate-limiting-advanced:
    - Fixed an issue where the runtime failed due to `sync_rate` not being set if the `strategy` was `local`.
- response-ratelimiting:
    - Fixed an issue where usage headers that were supposed to be sent to the upstream were lost instead.
- service-protection:
    - Enhanced robustness for user misconfigurations. The following use cases are now handled:
        - RLA and service-protection are configured on the same service.
        - There is no service but the service-protection plugin is enabled with global scope.
    - Fixed an issue where the runtime failed due to `sync_rate` not being set if the `strategy` was `local`.
- grpc-web and **grpc-gateway**: Fixed a bug where the `TE` (transfer-encoding) header would not be sent to the upstream gRPC servers when `grpc-web` or `grpc-gateweay` are in use.
- Fixed an issue where templates weren't being resolved correctly.
- The plugins now support nested fields.

Configuration

- Fixed an issue where the `db_resurrect_ttl` configuration didn't take effect.

Admin API

- Fixed error caused by duplicate `Content-Type`.
- Fixed an issue where Admin API Enterprise-only entities were not writable when a license expired but was still in the grace period.
- Fixed an issue where the "meta" field was not validated when creating or updating a portal developer.

PDK

- Users can now use a backslash to escape dots in logging plugins' `custom_fields_by_lua` key strings, preventing dots from creating nested tables.

Default

- Created connection pools for each `host`, `port`, `username`, `ssl` combination to fix the following issues:

Kong Manager

- Fixed an issue where creating `jwt-credential` with special algorithms (`PS256`, `PS384`, `PS512`, and `EdDSA`) couldn't populate `rsa_public_key` in the Kong Manager.

ve the values of some fields (`client certificate`, `tags`, `timeouts`, and `host_header`, etc) in the Kong

lated incorrectly.

er some circumstances.

D

C

tains upgraded dependencies and a new interface for validating expressions.

XML_ResumeParser function caused by XML_StopParser stopping an uninitialized parser.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bumped OpenResty from 1.25.3.2 to 1.27.1.1.
- Bumped OpenSSL to 3.4.1 in Core dependencies.
- Bumped PCRE2 from 10.44 to 10.45 (https://github.com/PCRE2Project/pcre2/blob/pcre2-10.45/NEWS).
- Bumped Snappy Library from 1.2.0 to 1.2.1.
- Bumped libxml2 from 2.12.9 to 2.12.10.

Default

- Bumped `ngx_wasm_module` to `a376e67ce02c916304cc9b9ef25a540865ee6740`

CLI Command

- Updated included debug tools: `curl` to 8.12.1, the Mozilla CA Certificate Store to 2025-02-25, and `nghttp2` to 1.65.0.

Performance

Core

- Improved performance of trace ID size lookup.
- Reduced the LMDB storage space by optimizing the key format.

PDK

- Refined PDK usage for better performance.

Plugin

- oas-validation:
  - Improved performance on OpenAPI 3.0.
- openid-connect:
  - Removed issuer discovery from schema to improve performance upon plugin initialization or updating. The issuer discovery will only be triggerd by client requests.

Known Issues

Plugin

- **Confluent Consume and Kafka Consume plugins**:
  - An error message appears in the logs about a missing cluster name, even when the name is specified.
- **Vault Auth**:
  - The Vault Auth plugin doesn't clear its cache when incremental sync is turned on. This means that deleted secrets will remain in the cache, and can still be accessed by the plugin.
- ai-proxy:

eported as having zero length when using the AI Proxy plugin.

Consume plugin without authentication settings, but authentication must be configured for the plugin to

ication strategy is missing, the plugin will fail with a generic authentication error.

RM64 Kong Gateway Docker images:

ync=on), Kong Gateway can't apply configuration deltas to the stream subsystem. This issue affects
nabled alongside stream proxying (stream_listen). As a workaround for this issue, set

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- A bug originating from the underlying OpenResty platform affects connection pooling in certain circumstances. This issue impacts transaction handling and negatively impacts performance, resulting in reduced request rates and increased latencies. Versions 3.10.0.0 and 3.10.0.1 are both affected. To resolve this regression, please upgrade to Kong Gateway 3.10.0.2 or later.

3.9.1.2

Release date 2025/07/07

Feature

PDK

- Added a new `kong.request.get_raw_forwarded_path()` function for returning non-normalized `forwarded_path`.

Core

- Added support for reporting whether `new_dns_client` was enabled in anonymous reports.

Bugfix

Core

- Added a new configuration, `route_match_calculation`, which included bug fixes for the `traditional_compatible` and the `expressions` router flavors. The default value `original` retained the current `3.x` router matching behavior without changes. The value `strict` enforced the router matching behavior with corrected calculation as follows:
  1. The weight (priority point) of a route was incorrectly downgraded.
  2. A route of HTTPS-only protocol with SNIs incorrectly captured requests over HTTP routes.
  3. A route of TLS-only protocol with SNIs incorrectly captured requests over TCP or UDP routes.
- -> Fixed an issue where newly spawned workers used outdated router data because the `init` phase was skipped.
- Fixed an issue where users encountered duplicate key violation errors on `rbac_role_entities_pkey` while using the `deck sync` command.
- Fixed an issue where adding the `hash_subject` and `store_metadata` fields to the `portal_session_conf` in the Dev Portal was not working as expected.

Plugin

- ldap-auth-advanced:
  - Fixed an issue where binary string was truncated at the first null character.
- mocking:
  - Fixed an issue where an empty array set in the mocking plugin returned an object type instead of an array type in the response body.
- oas-validation:
  - Fixed an issue where the plugin throws an error for responses without a Content-Type header.
- openid-connect:
  - Fixed an issue where caused IdP to report invalid redirect_uri errors when `config.redirect_uri` was not configured and the uri path contained spaces.
- Fixed an issue where AI Proxy and AI Proxy Advanced would use corrupted plugin config.

P

dlock with Postgres.

D

C

3

R

E

P

- **AI Plugins**:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed issue of template not being resolved correctly and supported nested fields.

- ai-proxy:
    - Fixed preserve mode.
- ai-semantic-cache:
    - Fixed issue of SSE body may have extra trailing in some cases.
- app-dynamics:
    - Fixed segmentation fault caused by missing destructor call on process exit.

Core

- Fixed an issue where modifying x-forwarded header before access phase may not take effect
- Fixed an issue where DNS answers with TTL=0 were incorrectly cached indefinitely in the new DNS client.
- Fixed an issue where Konnect analytics were missing for Kong AI Gateway.

3.9.1.0

Release date 2025/03/11

Feature

Plugin

- session:
    - Added two boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) to store session's metadata in the database.

Core

- Added a new feature to invalidate the admin's or the developer's related session while changing the password.

Bugfix

Core

- Fixed an issue where A mismatch between If-Match in request and ETag in response would result in bad case in the response phase.
- **Vault**: Updated the AWS Vault supported regions list to the latest.
- Added support for the new Ollama streaming content type in AI driver.

Plugin

- ai-proxy:
    - Fixed Gemini streaming responses getting truncated and/or missing tokens.
    - Fixed incorrect error thrown when trying to log streaming responses.

Bedrock and Gemini providers.

onfigure the plugin when using DB-less mode.

3

R

P

P

to "ai-proxy" to align with previous versions.

    - Fixed a bug in the Azure provider where `model.options.upstream_path` overrides would always return 404.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed a bug where Azure streaming responses would be missing individual tokens.
    - Fixed a bug where response streaming in Gemini and Bedrock providers was returning whole chat responses in one chunk.
    - Fixed a bug where multimodal requests (in OpenAI format) would not transform properly, when using the Gemini provider.
- grpc-web and **grpc-gateway**: Fixed a bug where the `TE` (transfer-encoding) header would not be sent to the upstream gRPC servers when `grpc-web` or `grpc-gateway` are in use.

Core

- Fixed an issue where consistent hashing did not correctly handle hyphenated-Pascal-case headers, leading to uneven distribution of requests across upstream targets.
- Fixed an issue that certificate entity configured with vault reference may not get refreshed on time when initial with an invalid string.

Dependency

Core

- Bumped libexpat from 2.6.2 to 2.6.4 to fix a crash in the XML_ResumeParser function caused by XML_StopParser stopping an uninitialized parser.
- Bumped lua-kong-nginx-module from 0.13.1 to 0.13.2.

3.9.0.0

Release date 2024/12/12

Deprecation

Core

- `node_id` in configuration has been deprecated.

Plugin

- Fix an issue where running the “kong migration” command will fail when upgrading to 3.8 version, which is caused by an incomplete Redis configuration related SQL.

Feature

Plugin

- ai-proxy-advanced:
    - Added support for streaming responses to the AI Proxy Advanced plugin.
- ai-proxy:
    - Disabled HTTP/2 ALPN handshake for connections on routes configured with AI-proxy.
- ai-rate-limiting-advanced:
    - Added support for Huggingface provider to the AI Rate Limiting Advanced plugin.
- ai-semantic-cache:

d tool role prompts from the input text.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

orts blocking requests based on regex patterns.

ode

roxy_authorization to be referenceable.

aders config option, allowing you to pass client headers as introspection POST body arguments.

maximal capacity to 6000

    - Added support for Proxy-Wasm metrics.

- rate-limiting-advanced:
    - Added a new configuration field `lock_dictionary_name` to support specifying an independent shared memory for storing locks.
    - Added support for authentication from Kong Gateway to Envoy Proxy.
    - Added support for combining multiple identifier items with the new configuration field `compound_identifier`.
- redirect:
    - Add a new plugin to redirect requests to another location
- service-protection:
    - Implemented a new plugin to protect services with request rate limiting.
- ai-semantic-cache, **ai-semantic-prompt-guard**, **ai-proxy-advanced**: Made the `embeddings.model.name` config field a free text entry, enabling use of a self-hosted (or otherwise compatible) model.

Clustering

- Added a remote procedure call (RPC) framework for Hybrid mode deployments.

Core

- **Core**: Added Ada dependency - WHATWG-compliant and fast URL parser.
- Addded a new LLM driver for interfacing with the Hugging Face inference API. The driver supports both serverless and dedicated LLM instances hosted by Hugging Face for conversational and text generation tasks.
- **Core**: Added `tls.disable_http2_alpn()` function needed patch for disabling HTTP/2 ALPN when tls handshake.
- Improved the output of the request debugger:
    - Now the resolution of field `total_time` is microseconds.
    - A new field `total_time_without_upstream` on the top level shows the latency only introduced by Kong.
- **proxy-wasm**: Added support for Wasm filters to be configured via the /plugins admin API
- Added a new feature for Kong Manager that supports multiple domains, enabling dynamic cross-origin access for Admin API requests.
- Added a configuration parameter `admin_gui_auth_login_attempts_ttl` (default to `604800`) to allow users to specify a custom duration to wait before they can try login again if they have exceeded the maximum login attempts. This is only meaningful when `admin_gui_auth_login_attempts` is a positive number.
- let the embedding driver cache the embeddings for a given model in the current request
- Added an option for GitHub Actions to build nginx/OpenResty with debug symbols.

Admin API

- **Admin API**: Added support for official YAML media-type (application/yaml) to /config endpoint.
- Added the ability to remove the consumer list from the return value for consumer groups Admin API `/consumer_groups/:consumer_groups` when `list_consumers=false`.
- Entity counts in `/license/report` were retrieved with `select count` instead of `workspace_entity_counters` table on db mode.
- Entity counts in `/workspaces?counter` and `/workspace/<workspace>/meta` were retrieved with `select count` instead of `workspace_entity_counters` table on

ss mode.

eta could retrieve entity counts under db-less mode.

e field of an admin via the Admin API and Kong Manager.

C

/ready endpoint return `503 Service Unavailable` response.

P

) to PDK.

the tracing PDK

K

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Kong Manager will now show a more friendly error message when failing to delete a service.

Bugfix

Plugin

- **OAS**:
  - Fixed an issue where the spec could not be located if the `Content-Type` in the request/response body included parameters (e.g., `application/json; charset=utf8`), while the openapi specification defined in `api_spec` did not include parameters.
- **ai-transformers**:
  - Fixed a bug where the correct LLM error message was not propagated to the caller.
- ai-prompt-guard:
  - Fixed an issue where the *ai-prompt-guard* plugin could fail when handling requests with multiple models.
- ai-proxy-advanced:
  - Fixed an issue where lowest-usage and lowest-latency strategy did not update data points correctly.
- ai-proxy:
  - Fixed a bug where tools (function) calls to Anthropic would return empty results.
  - Fixed a bug where tools (function) calls to Bedrock would return empty results.
  - Fixed a bug where Bedrock Guardrail config was ignored.
  - Fixed a bug where tools (function) calls to Cohere would return empty results.
  - Fixed a bug where Gemini provider would return an error if content safety failed in AI Proxy.
  - Fixed a bug where tools (function) calls to Gemini (or via Vertex) would return empty results.
  - Fixed an issue where AI Transformer plugins always returned a 404 error when using 'Google One' Gemini subscriptions.
  - Fixed issue where multi-modal requests is blocked on azure provider.
- ai-rate-limiting-advanced:
  - Updated the error message for rate limit exceeded to include AI-related information.
  - Fixed an issue where the plugin yielded an error when incrementing the rate limit counters in non-yieldable phases.
  - Fixed an issue where the plugin may fail to authenticate to Redis correctly with vault-referenced redis configuration.
- ai-semantic-cache:
  - Fixed the exact matching to catch everything including embeddings.
  - Fixed an issue where the ai-semantic-cache plugin put the wrong type value in the metrics when using the prometheus plugin.
  - Fixed an issue where the plugin failed when handling requests with multiple models.
- ai-semantic-prompt-guard:
  - Fixed an issue where requests with multiple models caused failures.
- app-dynamics:

stream, service, route and consumer was missing in the AppDynamics plugin.

d internal server error when the `multiValueHeaders` is null.

dated from the control plane but not updated in the degraphql router on the data plane.

take effect on invalid non-admin requests.

icate to Redis correctly with vault-referenced redis configuration.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Fixed an issue where the length counting of escape sequences, non-ASCII characters, and object entry names in JSON Strings was incorrect; now using UTF-8 character count instead of bytes.
    - Fixed an issue where certain default parameter values were incorrectly interpreted as 0 in some environments (e.g., ARM64-based):
        - `max_container_depth`
        - `max_object_entry_count`
        - `max_object_entry_name_length`
        - `max_array_element_count`
        - `max_string_value_length`
- jwe-decrypt:
    - Fixed an issue where an unnecessary warn log was printed.
- jwt:
    - ensure `rsa_public_key` isn't base64-decoded.
- kafka-log:
    - Fixed an issue where the plugin cannot function correctly when it is configured in a non-default workspace with certificate_id.
    - Reduced noisy logs from kafka-log and counters.
- key-auth:
    - Fixed to retain order of query arguments when hiding the credentials.
- oas-validation:
    - Fixed an issue where the error message was omitted if `notify_only_request_body_validation_failure` or `notify_only_response_body_validation_failure` was set to `false`.
    - Fixed an issue where the `include_base_path` did not work when multiple servers were provided.
- openid-connect:
    - Fixed an `500` error caused by JSON `null` from the request body when parsing bearer tokens or client IDs.
    - Fixed an issue where the configured Redis database was ignored.
    - Fixed an issue where the `token_cache_key_include_scope` feature was not considering scopes defined via `config.scopes` to generate the cache key.
- rate-limiting-advanced:
    - Fixed an issue where counters of the overriding consumer groups didn't fetched when the window_size is different and the workspace is non-default.
    - Fixed an issue where a warn log was printed when event_hooks was disabled.
    - Fixed an issue where multiple plugin instances sharing the same namespace enforced consumer groups and different `window_sizes` were used in the consumer group overriding configs, then the rate limiting of some consumer groups would fall back to local strategy. Now every plugin instance sharing the same namespace can set different `window_size`.
    - Fixed an issue where the plugin may fail to authenticate to Redis correctly with vault-referenced redis configuration.
    - Fixed an issue where RLA stores long expiration time items cause `no memory` errors.
- rate-limiting:
    - Fix a bug where the return values from `get_redis_connection()` are mistaken.
    - Fixed an issue that caused an HTTP 500 error when `hide_client_headers` is set to `true` and the request exceeds the rate limit.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

fining an object parameter with form style and exploded.

vided models

n issue where stale plugin config was not updated in dbless and hybrid mode.

abort in stream mode when another plugin enable the buffering proxy mode.

an issue where Azure Managed Identity did not work for AI Transformer Plugins.

ections.

Planes.

me role and custom STS endpoint features backport.

- Fixed a connection leak issue where the websocket connection was not closed promptly during reconnection.

Admin API

- Fix for querying admin API entities with empty tags
- Fixed an issue where nested parameters can not be parsed correctly when using form-urlencoded requests.
- Fixed the issue where the entities' counter was not displayed in certain cases when they were empty.

Core

- Fixed a bug where the health checker could fail to initialize in rare cases.
- Fix to always pass `ngx.ctx` to `log_init_worker_errors` as otherwise it may runtime crash.
- **Loggly**: Fixed an issue where `/bin/hostname` missing caused an error warning on startup.
- **Core**: Fixed an issue where `ngx.balancer.recreate_request` API does not refresh body buffer when `ngx.req.set_body_data` is used in balancer phase
- Fixed an issue where the workspace id was not included in the plugin config in the plugins iterator.
- Fixed a 500 error triggered by unhandled nil fields during schema validation.
- **Vault**: Fixed an issue where array-like configuration fields cannot contain vault reference.
- **Vault**: Fixed an issue where updating a vault entity in a non-default workspace will not take effect.
- **Vault**: Fixed an issue where vault reference in kong configuration cannot be dereferenced when both http and stream subsystems are enabled.
- **proxy-wasm:** Added a check that prevents Kong from starting when the database contains invalid Wasm filters.
- Fixed an issue where the `kong.request.enable_buffering` can not be used when downstream uses HTTP/2.
- Fixed an issue where paginated results of `audit_requests` fetched via the `next` field were incorrect when `before` and `after` filters were applied.
- Fixed an issue where event_hooks added during runtime didn't function until restart.
- Fixed an issue where using Hashicorp Vault AppRole authentication with a secret ID file would fail to read the secret ID.
- Fixed an issue where RBAC authorization could be enabled via `enforce_rbac` in DB-less mode. RBAC authorization should be disabled in DB-less mode.
- Fixed an issue where massive routes insertion causes crashing and 500.

PDK

- Line up the `kong.log.inspect` function to log at `notice` level as documented in the PDK documentation (used to be `debug`).
- Fixed an issue where the `retries` error message incorrectly referred to the port.

Kong Manager

- Fixed an issue where the content in the header and footer were not center aligned when enabled.
- Fixed an issue where Kong Manager was not displaying the overview page when there is a workspace with the name "portal" (case-insensitive).
- Fixed an issue where Kong Manager was not redirecting users to the previous page after cancelling the plugin creation.
- Fixed an issue where the username does not allow special characters.

Dependency

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

to UBI 9.

fix the upstream cert chain issue and enable the new API for retrieving SSL pointer.

prefix generating

usage.

ng length calculation and added support for `null` in `enum` types.

e 1.

eter to allow overriding page size from caller side.

- Bumped `ngx_wasm_module` to `9150e403a0f1d80755ce66c88c3ddecd0eb5e25d`

- Bumped `Wasmtime` version to `26.0.0`
- Bumped OpenSSL to 3.2.3, to fix unbounded memory growth with session handling in TLSv1.3 and other CVEs
- Bumped `kong-redis-cluster` to `1.5.5`.
    1. Currently, the timeout for acquiring a lock is fixed to `5s`. We added a new option `lock_timeout` to make it configurable.
    2. The lock timeout parameter was incorrectly set to `time_out = 0`. We fix it to `timeout = 0`. This would improve perf as there is no need for each instance to refresh the slots.
    3. Returned detailed error message to downstream component (e.g. Kong Gateway) for better debuggability.
- Bumped lua-resty-azure to 1.6.1 to fix a GET request build issue
- Added Ubuntu 24.04 (Noble Numbat) FIPS packages and image.

Known Issues

Core

- The Brotli module is missing from all of the following ARM64 Kong Gateway Docker images:
    - RHEL 9
    - Debian 12
    - Amazon Linux 2
    - Amazon Linux 2023

3.8.1.2

Release date 2025/07/02

Feature

PDK

- Added a new `kong.request.get_raw_forwarded_path()` function for returning non-normalized `forwarded_path`.

Core

- Added support for reporting whether `new_dns_client` was enabled in anonymous reports.

Bugfix

Core

- Added a new configuration, `route_match_calculation`, which included bug fixes for the `traditional_compatible` and the `expressions` router flavors. The default value `original` retained the current `3.x` router matching behavior without changes. The value `strict` enforced the router matching behavior with corrected calculation as follows:
    1. The weight (priority point) of a route was incorrectly downgraded.
    2. A route of HTTPS-only protocol with SNIs incorrectly captured requests over HTTP routes.
    3. A route of TLS-only protocol with SNIs incorrectly captured requests over TCP or UDP routes.
- Fixed an issue where newly spawned workers used outdated router data because the `init` phase was skipped.
- Fixed an issue where users encountered duplicate key violation errors on `rbac_role_entities_pkey` while using the `deck sync` command.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

king plugin returned an object type instead of an array type in the response body.

responses without a Content-Type header.

request body when parsing bearer tokens or client IDs.

edirect_uri errors when `config.redirect_uri` was not configured and the uri path contained spaces.

ong expiration time items cause `no memory` errors.

P

- Fixed an issue where rate-limiting library could cause deadlock with Postgres.

Dependency

Core

- Bumped luarocks from 3.11.1 to 3.12.1.

3.8.1.1

Release date 2025/04/10

Deprecation

Plugin

- Fix an issue where running the “kong migration” command will fail when upgrading to 3.8 version, which is caused by an incomplete Redis configuration related SQL.

Feature

Plugin

- session:
  - Added two boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) to store session’s metadata in the database.

Core

- Added an option for GitHub Actions to build nginx/OpenResty with debug symbols.
- Added a new feature to invalidate the admin’s or the developer’s related session while changing the password.

Bugfix

Core

- Fixed an issue where consistent hashing did not correctly handle hyphenated-Pascal-case headers, leading to uneven distribution of requests across upstream targets.
- Fixed an issue that certificate entity configured with vault reference may not get refreshed on time when initial with an invalid string.
- Fixed an issue where A mismatch between If-Match in request and ETag in response would result in bad case in the response phase.
- **Vault**: Updated the AWS Vault supported regions list to the latest.
- Fixed an issue where adding the `hash_subject` and `store_metadata` fields to the `portal_session_conf` in the Dev Portal was not working as expected.

Plugin

- app-dynamics:
  - Fixed segmentation fault caused by missing destructor call on process exit.
- ldap-auth-advanced:
  - Fixed an issue where binary string was truncated at the first null character.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

che-advanced.migrations module was not being loaded when upgrading to 3.8.x.y. This issue was
redis.timeout and redis.connect_timeout are set to different values.

iting-advanced.migrations module was not being loaded when upgrading to 3.8.x.y. This issue was
redis.timeout and redis.connect_timeout are set to different values.

XML_ResumeParser function caused by XML_StopParser stopping an uninitialized parser.

Release date 2024/11/04

Feature

Plugin

- prometheus:
  - Bumped KONG_LATENCY_BUCKETS bucket's maximal capacity to 6000

Bugfix

Plugin

- **ai-transformers**:
  - Fixed a bug where the correct LLM error message was not propagated to the caller.
- ai-proxy-advanced:
  - Fixed an issue where lowest-usage and lowest-latency strategy did not update data points correctly.
- ai-proxy:
  - Fixed an issue where AI Transformer plugins always returned a 404 error when using 'Google One' Gemini subscriptions.
  - Fixed issue where multi-modal requests is blocked on azure provider.
- ai-rate-limiting-advanced:
  - Fixed an issue where the plugin yielded an error when incrementing the rate limit counters in non-yieldable phases.
- ai-semantic-cache:
  - Fixed an issue where the ai-semantic-cache plugin put the wrong type value in the metrics when using the prometheus plugin.
- degraphql:
  - Fixed an issue where the degraphql routes were updated from the control plane but not updated in the degraphql router on the data plane.
- json-threat-protection:
  - Fixed an issue where the length counting of escape sequences, non-ASCII characters, and object entry names in JSON Strings was incorrect; now using UTF-8 character count instead of bytes.
  - Fixed an issue where certain default parameter values were incorrectly interpreted as 0 in some environments (e.g., ARM64-based):
    - `max_container_depth`
    - `max_object_entry_count`
    - `max_object_entry_name_length`
    - `max_array_element_count`
    - `max_string_value_length`
- rate-limiting-advanced:
  - Fixed an issue where a warn log was printed when event_hooks was disabled.
- rate-limiting:
  - Fixed an issue that caused an HTTP 500 error when `hide_client_headers` is set to `true` and the request exceeds the rate limit.

vided models

n issue where stale plugin config was not updated in dbless and hybrid mode.

abort in stream mode when another plugin enable the buffering proxy mode.

A

C

on-default workspace will not take effect.

C

me role and custom STS endpoint features backport.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Kong Manager

- Fixed an issue where text was not centered in custom banners.
- Fixed an issue where a workspace named ‘portal’, but with different case letters, does not render the correct overview page.

Dependency

Core

- Bumped lua-kong-nginx-module from 0.11.0 to 0.11.1 to fix an issue where the upstream cert chain wasn’t properly set
- Bumped lua-resty-aws to 1.5.4, to fix a bug inside region prefix generating
- Bumped lua-resty-azure to 1.6.1 to fix a GET request build issue

3.8.0.0

Release date 2024/09/11

Deprecation

Core

- Debian 10, CentOS 7, and RHEL 7 reached their End of Life (EOL) dates on June 30, 2024. As of version 3.8.0.0 onward, Kong is not building installation packages or Docker images for these operating systems. Kong is no longer providing official support for any Kong version running on these systems.

PDK

- The shared configuration for Redis `kong/enterprise_edition/redis/init.lua` was deprecated in favor of `kong/enterprise_edition/tools/redis/v2/init.lua`

Plugin

- ai-rate-limiting-advanced:
  - Switched to sentinel_nodes and cluster_nodes for redis configuration.
  - Deprecated timeout config field in redis config in favor of connect_/send_/read_timeout (timeout field will be removed in 4.0).
- graphql-proxy-cache-advanced:
  - Switched to sentinel_nodes and cluster_nodes for redis configuration.
  - Deprecated timeout config field in redis config in favor of connect_/send_/read_timeout (timeout field will be removed in 4.0).
- graphql-rate-limiting-advanced:
  - Deprecated timeout config field in redis config in favor of connect_/send_/read_timeout (timeout field will be removed in 4.0).
  - Switched to sentinel_nodes and cluster_nodes for redis configuration.
- openid-connect:
  - Standardized Redis configuration across plugins. The Redis configuration now follows a common schema shared with other plugins.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

avor of connect_/send_/read_timeout (timeout field will be removed in 4.0).

edis configuration.

avor of connect_/send_/read_timeout (timeout field will be removed in 4.0).

edis configuration.

he Redis configuration now follows a common schema shared with other plugins.

    - allow AI plugin to read request from buffered file
- acl:
    - Added a new config `always_use_authenticated_groups` to support using authenticated groups even when an authenticated consumer already exists.
- ai-prompt-guard:
    - add `match_all_roles` option to allow match all roles in addition to `user`.
- ai-proxy-advanced:
    - Added the `ai-proxy-advanced` plugin that supports advanced load balancing between LLM services.
- ai-proxy:
    - Add `allow_override` option to allow overriding the upstream model auth parameter or header from the caller's request.
    - Allowed mistral provider to use mistral.ai managed service by omitting upstream_url
    - Added a new response header X-Kong-LLM-Model that displays the name of the language model used in the AI-Proxy plugin.
- ai-rate-limiting-advanced:
    - Add the cost strategy to AI rate Limiting plugin.
    - Added the `bedrock` and `gemini` providers to the providers list in the `ai-rate-limiting-advanced` plugin.
    - Add the stats when reaching limit and exiting AI rate Limiting plugin.
    - Added Redis cluster_max_redirections configuration option.
- ai-semantic-cache:
    - Introduced AI Semantic Caching plugin, enabling you to configure an embeddings-based caching system for Large Language Model responses.
- ai-semantic-prompt-guard:
    - Added the `ai-semantic-prompt-guard` plugin that supports semantic similarity-based prompt guarding.
- app-dynamics:
    - Added new ANALYTICS_ENABLE flag and collected more snapshot userdata in runtime.
- aws-lambda:
    - A new configuration field `empty_arrays_mode` is now added to control whether Kong should send `[]` empty arrays (returned by Lambda function) as `[]` empty arrays or `{}` empty objects in JSON responses.`
    - Added support for a configurable STS endpoint with the new configuration field `aws_sts_endpoint_url`.
- confluent:
    - Added the `confluent` plugin which allows to interface with Confluent.
- graphql-proxy-cache-advanced:
    - Added Redis cluster_max_redirections configuration option.
- graphql-rate-limiting-advanced:
    - Added Redis cluster_max_redirections configuration option.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

thentication.

ON nesting depth, array elements, object entries, key length, and string length. Logs or terminates violating

endpoint in dbless mode.

sented in long form length

alue when the path parameter name contains hyphen characters.

  - Added `claims_forbidden` property to restrict access.
  - Added support for redis cache for introspection result with new fields `cluster_cache_strategy` and `cluster_cache_redis`. When configured, the plugin will share the tokens introspection responses cache across nodes configured to use the same Redis Database.
- opentelemetry:
  - Added support for OpenTelemetry formatted logs.
- proxy-cache-advanced:
  - Added Redis cluster_max_redirections configuration option.
- rate-limiting-advanced:
  - Added Redis cluster_max_redirections configuration option.
- request-transformer:
  - Fixed an issue where renamed query parameters, url-encoded body parameters, and json body parameters were not handled properly when target name is the same as the source name in the request.
- standard-webhooks:
  - Added standard webhooks plugin.
- upstream-oauth:
  - Added the Upstream OAuth plugin, enabling Kong to obtain an OAuth2 token to consume an upstream API.
- AI plugins: retrieved latency data and pushed it to logs and metrics.
- Kong AI Gateway (AI Proxy and associated plugin family) now supports all AWS Bedrock “Converse API” models.
- Kong AI Gateway (AI Proxy and associated plugin family) now supports the Google Gemini “chat” (generateContent) interface.
- Added support for json_body rename in response-transformer plugin

Core

- **prometheus**: Added `ai_requests_total`, `ai_cost_total` and `ai_tokens_total` metrics in the Prometheus plugin to start counting AI usage.
- Added a new configuration `concurrency_limit`(integer, default to 1) for Queue to specify the number of delivery timers. Note that setting `concurrency_limit` to `-1` means no limit at all, and each HTTP log entry would create an individual timer for sending.
- Append gateway info to upstream `Via` header like `1.1 kong/3.8.0`, and optionally to response `Via` header if it is present in the `headers` config of “kong.conf”, like `2 kong/3.8.0`, according to `RFC7230` and `RFC9110`.
- Starting from this version, a new DNS client library has been implemented and added into Kong, which is disabled by default. The new DNS client library has the following changes - Introduced global caching for DNS records across workers, significantly reducing the query load on DNS servers. - Introduced observable statistics for the new DNS client, and a new Status API `/status/dns` to retrieve them. - Simplified the logic and make it more standardized
- **analytics**: send AI analytics about latency and caching to Konnect.
- **analytics**: Added support for also sending cache data of AI analytics to Konnect
- Added connection support via Redis Proxy (e.g. Envoy Redis proxy or Twemproxy) via configuration field `connection_is_proxied`.
- Added support for AWS IAM role assuming in AWS IAM Database Authentication, with new configuration fields: “pg_iam_auth_assume_role_arn”, “pg_iam_auth_role_session_name”, “pg_ro_iam_auth_assume_role_arn”, and “pg_ro_iam_auth_role_session_name.”
- Added keyring encryption support to license database entity payloads.

IAM Authentication, with new configuration fields: `pg_iam_auth_sts_endpoint_url` and

Vault. This can either be configured by `vault_aws_sts_endpoint_url` as a global configuration, or

`sword_attempts` (default value `0`) and `admin_gui_auth_change_password_ttl` (default value `86400`), to

LI tool.

A

ration via the Admin API

P

`max_allowed_file_size` is `0`, `get_raw_body` will return the entire body, but the size of this body will still

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- extend kong.request.get_body and kong.request.get_raw_body to read from buffered file
- Added a new PDK module `kong.telemetry` and function: `kong.telemetry.log` to generate log entries to be reported via the OpenTelemetry plugin.

Configuration

- Configure Wasmtime module cache when Wasm is enabled

Kong Manager

- Kong Manager will now show input boxes that allow optionally creating SNIs while creating a certificate.
- While deleting a workspace, Kong Manager will now list admins that prevent the operation.
- Kong Manager will now show scoping entities as links in the plugin detail page.
- Added UI components for building the vault reference easily while configuring referenceable fields for plugins.

Bugfix

Plugin

- **AI Plugins**:
  - Fixed an issue for multi-modal inputs are not properly validated and calculated.
- **AI-Transformers**:
  - Fixed a bug where cloud identity authentication was not used in `ai-request-transformer` and `ai-response-transformer` plugins.
- **OpenTelemetry / Zipkin**:
  - remove redundant deprecation warnings
- acme:
  - Fixed an issue of DP reporting that deprecated config fields are used when configuration from CP is pushed
  - Fixed an issue where username and password were not accepted as valid authentication methods.
- ai-prompt-guard:
  - Fixed an issue when `allow_all_conversation_history` is set to false, the first user request is selected instead of the last one.
- ai-proxy:
  - Fixed a bug where certain Azure models would return partial tokens/words when in response-streaming mode.
  - Fixed a bug where Cohere and Anthropic providers don't read the `model` parameter properly from the caller's request body.
  - Fixed a bug where using "OpenAI Function" inference requests would log a request error, and then hang until timeout.
  - Fixed a bug where AI Proxy would still allow callers to specify their own model, ignoring the plugin-configured model name.
  - Fixed a bug where AI Proxy would not take precedence of the plugin's configured model tuning options, over those in the user's LLM request.
  - Fixed a bug where setting OpenAI SDK model parameter "null" caused analytics to not be written to the logging plugin(s).
  - Fixed issue when response is gzipped even if client doesn't accept.

set data on the class instead of the instance

issing passing window to shm

tion for calculating latencies when no embeddings, add deep copy for the request table and fix countback.

ultiValueHeaders defined in proxy integration and legacy empty_arrays_mode.

e request payload when `awsgateway_compatible` is enabled.

kong versions (before 3.6)

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Fixed an issue where the plugin would not work if we explicitly set the `generator` to `null`.

- cors:
  - Fixed an issue where the `Access-Control-Allow-Origin` header was not sent when `conf.origins` has multiple entries but includes `*`.
- degraphql:
  - Fixed an issue where multiple parameter types were not handled correctly when converting query parameters.
- grpc-gateway:
  - When there is a JSON decoding error, respond with status 400 and error information in the body instead of status 500.
- hmac-auth:
  - Add WWW-Authenticate headers to 401 responses.
- http-log:
  - Fix an issue where the plugin doesn't include port information in the HTTP host header when sending requests to the log server.
- jwt:
  - Add WWW-Authenticate headers to 401 responses.
- key-auth-enc:
  - Added WWW-Authenticate headers to all 401 responses.
- key-auth:
  - Fix an issue of realm field not recognized for older kong versions (before 3.7)
- ldap-auth-advanced:
  - Added WWW-Authenticate headers to all 401 response.
- ldap-auth:
  - Add WWW-Authenticate headers to all 401 responses.
- oas-validation:
  - Fixed an issue where parameter serialization does not behave the same as in the OpenAPI specification
  - Fixed a bug where the non-string primitive types passed via URL query were unexpectedly cast to string when OpenAPI spec is v3.1.0.
- oauth2-introspection:
  - Fixed an issue where the consumer's cache cannot be invalidated when oauth2-introspection uses `client_id` as `consumer_by`.
- oauth2:
  - Add WWW-Authenticate headers to all 401 responses and realm option.
- openid-connect:
  - Fixed a bug where anonymous consumers may be cached as nil under a certain condition.
  - Updated the rediscovery to use a short lifetime (5s) if the last discovery failed.
  - Fixed an issue where using_pseudo_issuer does not work when patching.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ng from below version 3.3 to 3.7.

as missing for the new configuration field `ai_metrics`.

abels count.

odated correctly when serving cached requests

updated correctly when serving cached responses.

    - Fixed an issue where if the `window_size` in the consumer group overriding config is different from the `window_size` in the default config, the rate limiting of that consumer group would fall back to local strategy.

- rate-limiting:
    - Fixed an issue of DP reporting that deprecated config fields are used when configuration from CP is pushed
- request-size-limiting:
    - Fixed an issue where the body size doesn't get checked when the request body is buffered to a temporary file.
- request-validator:
    - Fix an issue where the plugin may fail to handle requests when param_schema is $ref schema.
    - Added a new configuration field `content_type_parameter_validation` to determine whether to enable Content-Type parameters validation.
- response-ratelimiting:
    - Fixed an issue of DP reporting that deprecated config fields are used when configuration from CP is pushed
- statsd:
    - Fixed an issue where the exported workspace was always `default` when the workspace identifier was set to the workspace name.
- tls-metadata-headers:
    - Fixed an issue where intermediate certificates details were not added to request headers.
- Fixed certain AI plugins cannot be applied per consumer or per service.
- Fixed a bug where Azure Managed-Identity tokens would never rotate
  in case of a network failure when authenticating.
- Fixed a Redis schema issue where `connect_timeout`, `read_timeout`, `send_timeout` were reset to `null` if the deprecated `timeout` is `null`.
- rate-limiting-advanced Fixed an issue where the sync timer may stop working due to race condition.

Admin API

- Fixed an issue where validation of the certificate schema failed if the `snis` field was present in the request body.
- Fixed an issue where resetting the token was allowed while disabling rbac_token_enabled.
- The `application-registration` plugin will be hidden from `available_plugins` when the Dev Portal is disabled.
- Fixed an issue where the field `is_default` should be immutable when updating the rbac_roles.
- Fixed an issue where the license report returns 500 when non-required fields are not specified in the Lambda and Kafka plugins.
- Returns a detailed error message when failed to cascade delete a workspace caused by admins associated.

Clustering

- Fixed an issue where hybrid mode not working if the forward proxy password contains special character(#). Note that the `proxy_server` configuration parameter still needs to be url-encoded.

CLI Command

- Fixed an issue where some debug level error logs were not being displayed by the CLI.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

nses in declarative YAML.

C

tgres read-only database operations.

eplacement fields when both are specified.

ared when `ngx.send_header()` triggered `filter_finalize` openresty/lua-nginx-module#2323.

th new fields. If the new field contains null it allows for deprecated field to overwrite it if both are present

rror log is reported when 400 bad requests were received.

hen the first capture group is absent.

itional mode route When 'router_flavor' is configured as 'expressions'.

override the global level `proxy_ssl_verify`.

- Fixed an issue where the sni cache isn't invalidated when a sni is updated.
- The kong.logrotate configuration file will no longer be overwritten during upgrade. When upgrading, set the environment variable `DEBIAN_FRONTEND=noninteractive` on Debian/Ubuntu to avoid any interactive prompts and enable fully automatic upgrades.
- Fixed an issue where the Vault secret cache got refreshed during `resurrect_ttl` time and could not be fetched by other workers.
- Error logs during Vault secret rotation are now logged at the `notice` level instead of `warn`.
- fix a bug that the `host_header` attribute of upstream entity can not be set correctly in requests to upstream as Host header when retries to upstream happen.
- **AI-proxy**: A configuration validation is added to prevent from enabling `log_statistics` upon providers not supporting statistics. Accordingly, the default of `log_statistics` is changed from `true` to `false`, and a database migration is added as well for disabling `log_statistics` if it has already been enabled upon unsupported providers.
- Moved internal Unix sockets to a subdirectory (`sockets`) of the Kong prefix.
- Changed the behaviour of shorthand fields that are used to describe deprecated fields. If both fields are sent in the request and their values mismatch - the request will be rejected.
- Reverted DNS client to original behaviour of ignoring ADDITIONAL SECTION in DNS responses.
- Shortened names of internal Unix sockets to avoid exceeding the socket name limit.
- Built-in RBAC roles for admins (`admin` under the default workspace and `workspace-admin` under non-default workspaces) now disallow CRUD actions to `/groups` and `/groups/*` endpoints.
- Fixed an issue where luarocks-admin was not available in /usr/local/bin.
- Fixed an issue where running Kong CLI commands with database configurations containing Hashicorp Vault references would fail to execute.
- Fixed an issue where the stale license expiry warning continued to be logged even if the license was updated.
- License expiry warnings are no longer logged and license info is removed from /metrics in Konnect.
- Fixed an issue where the CPs won't trigger a configuration push after a keyring recovery.

PDK

- **PDK**: Fixed a bug that log serializer will log `upstream_status` as nil in the requests that contains subrequest
- **Vault**: Reference ending with slash when parsed should not return a key.
- Fixed an issue that pdk.log.serialize() will throw an error when JSON entity set by serialize_value contains json.null

Configuration

- Re-enabled the Lua DNS resolver from proxy-wasm by default.
- The behavior of the configuration option `analytics_flush_interval` has changed for saving memory resources by flushing analytics messages more frequently. It now controls the maximum time interval between two flushes of analytics messages to the configured backend, which means that if there are enough (less than `analytics_buffer_size_limit`) messages have already been buffered, the flush will happen before the configured interval. Previously, Kong always tries to flush messages after the configured interval, regardless of the number of messages in the buffer.
- Fixed an issue where `debug_listen` incorrectly used the SSL-related configuration of `status_listen`.

Kong Manager

- Fixed an issue where dynamic ordering was configurable for plugins scoped by consumers and/or consumer groups. These plugins does not support dynamic ordering

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ocal storage.

dresses fields for plugins that support Redis clusters.

as not correctly rendered.

active admin was updated.

D

C

ssword auth with redis.

regional endpoint.

n issue that was causing high memory usage

- Bumped lua-resty-lmdb to 1.4.3 to get fixes from the upstream (lmdb 0.9.33), which resolved numerous race conditions and fixed a cursor issue.
- Bumped lua-resty-openssl to 1.5.1 to fix some issues including a potential use-after-free issue.
- Bumped LuaRocks from 3.11.0 to 3.11.1
- Bumped `ngx_wasm_module` to `96b4e27e10c63b07ed40ea88a91c22f23981db35`
- Bumped OpenResty to 1.25.3.2 to improve the performance of the LuaJIT hash computation.
- Bumped PCRE2 to 10.44 to fix some bugs and tidy-up the release (nothing important)
- Bumped `Wasmtime` version to `25.0.1`
- Made the RPM package relocatable with the default prefix set to `/`.
- Introduced a yieldable JSON library `lua-resty-simdjson`, which would improve the latency significantly.
- Bumped `kong-lua-resty-kafka` to `0.20` to support TCP socket keepalive and allow client_id to be set for the kafka client.
- Bump lua-resty-jsonschema-rs to 0.1.5
- bump lua-resty-cookie to 0.3.0
- Bumped libxml2 to 2.12.9.
- Bumped libxslt to 1.1.42.
- Bumped `lua-resty-azure` to `1.6.0` to support more Azure authentication methods.
- Bumped luaexpat to 1.5.2.
- Bumped msgpack-c to 6.1.0.
- Bumped `kong-redis-cluster` to `1.5.4`, fixing the following issues.

  1. Fixed an issue where Kong Gateway cannot recover if partial or all pods were restared with new IPs in Kubernetes environment.
  2. Fixed a memory leak issue where master nodes cache expanded infinitely upon refresh.
  3. Fixed an issue where multiple cluster instances were accidently flushed.

Performance

Core

- Removed unnecessary DNS client initialization
- Improved latency performance when gzipping/gunzipping large data (such as CP/DP config data).
- Improved the performance of Konnect Analytics by fetching Rate Limiting context more efficiently.
- Improved the performance of Konnect Analytics by optimizing the buffering mechanism.

Plugin

- rate-limiting-advanced:
  - Improved that timer spikes do not occur when there is network instability with the central data store.

Feature

C

preventing the configuration from being updated to the latest version

K

C

RM64 Kong Gateway Docker images:

3

R

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Feature

Plugin

- session:
  - Added two boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) to store session's metadata in the database.

Core

- Added a new feature to invalidate the admin's or the developer's related session while changing the password.

Bugfix

Plugin

- app-dynamics:
  - Fixed segmentation fault caused by missing destructor call on process exit.
- ldap-auth-advanced:
  - Fixed an issue where binary string was truncated at the first null character.

Core

- **Vault**: Updated the AWS Vault supported regions list to the latest.
- Fixed an issue where adding the `hash_subject` and `store_metadata` fields to the `portal_session_conf` in the Dev Portal was not working as expected.

3.7.1.4

Release date 2025/02/25

Feature

Core

- Added an option for GitHub Actions to build nginx/OpenResty with debug symbols.

Bugfix

Core

- Fixed an issue that certificate entity configured with vault reference may not get refreshed on time when initial with an invalid string.

Dependency

Core

- Bumped libexpat from 2.6.2 to 2.6.4 to fix a crash in the XML_ResumeParser function caused by XML_StopParser stopping an uninitialized parser.
- Bumped lua-kong-nginx-module from 0.11.0 to 0.11.2

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Added support for a configurable STS endpoint for RDS IAM Authentication, with new configuration fields: `pg_iam_auth_sts_endpoint_url` and `pg_ro_iam_auth_sts_endpoint_url`.
- Added support for a configurable STS endpoint for AWS Vault. This can either be configured by `vault_aws_sts_endpoint_url` as a global configuration, or `sts_endpoint_url` on a custom AWS vault entity.

Bugfix

Plugin

- ai-proxy:
  - Fixed a bug where certain Azure models would return partial tokens/words when in response-streaming mode.
  - Fixed a bug where Cohere and Anthropic providers don't read the `model` parameter properly from the caller's request body.
  - Fixed a bug where using "OpenAI Function" inference requests would log a request error, and then hang until timeout.
  - Fixed a bug where AI Proxy would still allow callers to specify their own model, ignoring the plugin-configured model name.
  - Fixed a bug where AI Proxy would not take precedence of the plugin's configured model tuning options, over those in the user's LLM request.
  - Fixed a bug where setting OpenAI SDK model parameter "null" caused analytics to not be written to the logging plugin(s).
- rate-limiting-advanced:
  - Fixed an issue where if the `window_size` in the consumer group overriding config is different from the `window_size` in the default config, the rate limiting of that consumer group would fall back to local strategy.
  - Fixed an issue where the sync timer may stop working due to race condition.

Core

- The kong.logrotate configuration file will no longer be overwritten during upgrade. When upgrading, set the environment variable `DEBIAN_FRONTEND=noninteractive` on Debian/Ubuntu to avoid any interactive prompts and enable fully automatic upgrades.
- **Vault**: Fixed an issue where updating a vault entity in a non-default workspace will not take effect.
- Fixed an issue where the Vault secret cache got refreshed during `resurrect_ttl` time and could not be fetched by other workers.
- Moved internal Unix sockets to a subdirectory (`sockets`) of the Kong prefix.
- Shortened names of internal Unix sockets to avoid exceeding the socket name limit.
- Fixed an issue where luarocks-admin was not available in /usr/local/bin.

Dependency

Core

- Bumped lua-resty-aws to 1.5.3 to fix a bug related to STS regional endpoint.
- Bumped lua-resty-azure to 1.6.1 to fix a GET request build issue
- Made the RPM package relocatable with the default prefix set to `/`.

3.7.1.2

R

D

C

Life (EOL) dates on June 30, 2024. As of this patch, Kong is not building Kong Gateway 3.7.x installation
Kong is no longer providing official support for any Kong version running on these systems.

F

P

ow added to control whether Kong should send `[]` empty arrays (returned by Lambda function) as `[]`
es.`

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Bugfix

Kong Manager

- Fixed an issue where the Dev Portal documentation link was unavailable because the official documentation was removed after the 3.4.x.

Dependency

Core

- Bumped lua-resty-events to 0.3.0
- Bumped lua-resty-healthcheck to 3.1.0

3.7.1.1

Release date 2024/06/22

Bugfix

Core

- Reverted DNS client to original behaviour of ignoring ADDITIONAL SECTION in DNS responses.

3.7.1.0

Release date 2024/06/18

Bugfix

Plugin

- ai-proxy:
  - Resolved a bug where the object constructor would set data on the class instead of the instance
- ai-rate-limiting-advanced:
  - Edit the logic for the window ajustement and fix missing passing window to shm
- basic-auth:
  - Fix an issue of realm field not recognized for older kong versions (before 3.6)
- key-auth:
  - Fix an issue of realm field not recognized for older kong versions (before 3.7)
- openid-connect:
  - Fixed a bug where anonymous consumers may be cached as nil under a certain condition.
- request-validator:
  - Fix an issue where the plugin may fail to handle requests when param_schema is $ref schema.

rameter_validation to determine whether to enable Content-Type parameters validation.

itional mode route When 'router_flavor' is configured as 'expressions'.

y can not be set correctly in requests to upstream as Host header when retries to upstream happen.

workspace and workspace-admin under non-default workspaces) now disallow CRUD actions to /groups

ce active healthcheck timer usage.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bump lua-resty-jsonschema-rs to 0.1.5

Performance

Plugin

- rate-limiting-advanced:
  - Improved that timer spikes do not occur when there is network instability with the central data store.

3.7.0.0

Release date 2024/05/28

Breaking Change

Plugin

- ai-proxy:
  - To support the new messages API of `Anthropic`, the upstream path of the `Anthropic` for `llm/v1/chat` route type has changed from `/v1/complete` to `/v1/messages`.

Core

- "**Hashicorp Vault**: Starting from this version, a string fully made of spaces cannot be specified as the `role_id` or `secret_id` value in the Hashicorp Vault entity with Approle authentication method. "**Hashicorp Vault**: Starting from this version, at least one of `secret_id` and `secret_id_file` must be specified in the Hashicorp Vault entity with Approle authentication method.
- Removed the Granular Tracing feature, and configurations like `tracing = on` are not available any longer. You should now use OpenTelemetry Instrumentation instead.

Feature

Core

- Added `events:ai:response_tokens`, `events:ai:prompt_tokens` and `events:ai:requests` to the anonymous report to start counting AI usage
- Added support for debugging with EmmyLuaDebugger. This feature is a tech preview and not officially supported by Kong Inc. for now.
- Improved config handling when the CP runs with the router set to the `expressions` flavor:
  - If mixed config is detected and a lower DP is attached to the CP, no config will be sent at all
  - If the expression is invalid on the CP, no config will be sent at all
  - If the expression is invalid on a lower DP, it will be sent to the DP and DP validation will catch this and communicate back to the CP (this could result in partial config application)
- The route entity now supports the following fields when the `router_flavor` is `expressions`: `methods`, `hosts`, `paths`, `headers`, `snis`, `sources`, `destinations`, and `regex_priority`. The meaning of these fields are consistent with the traditional route entity.
- Kong Manager now supports creating and editing Expressions routes with an interactive in-browser editor with syntax highlighting and autocompletion features for Kong's Expressions language.
- Kong Manager now groups the parameters to provide a better user experience while configuring plugins. Meanwhile, several issues with the plugin form page were fixed.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

fields

des receive time/latency

to `true` for Server-Sent Event requests/responses.

C, LDAP), the source of an RBAC role will be stored in its `role_source` field, which enables the existing after IDP role mapping has changed. This also allows users to change a role's source between `local` and

ew options allow better control over the configuration of tracing headers propagation.

tor to enforce that all AI-Proxy requests must be introspected with the Azure Content Safety service.

different moderation categories, and reports audit results into the Kong log serializer for reporting purposes.

- ai-prompt-guard:
  - Increased the maximum length of regex expressions to 500 for the allow and deny parameters.
- ai-proxy:
  - Added support for streaming event-by-event responses back to the client on supported providers.
- graphql-proxy-cache-advanced:
  - Addded redis strategy support and `bypass_on_err` config for `graphql-proxy-cache-advanced` plugin.
- jwt-signer:
  - supports basic auth and mtls auth to external jwks services
  - The plugin now supports periodically rotating the jwks. For example, to autmatically rotate `access_token_jwks_uri`, you can set the config `access_token_jwks_uri_rotate_period`
  - The plugin now supports adding the original JWT(s) to the upstream request header by specifying the names of the upstream request header with `original_access_token_upstream_header` and `original_channel_token_upstream_header`. And `access_token_upstream_header`, `channel_token_upstream_header`, `original_access_token_upstream_header`, and `original_channel_token_upstream_header` should not have the same value.
- mocking:
  - Add the `custom_base_path` field to specifiy a custom base path. It will be used with the `deck file namespace` feature
- mtls-auth:
  - Add `default_consumer` option that allows a default consumer to be used when the client certificate is valid but does not match any existing consumers.
- oas-validation:
  - Add a new field `api_spec_encoded` to indicate whether the `api_spec` is URI-Encoded.
  - Add the `custom_base_path` field to specifiy a custom base path. It will be used with the `deck file namespace` feature
  - Supported OpenAPI Specification v3.1.0. The plugin now switches to a new JSONSchema validator when the specification version is v3.1.0.
- openid-connect:
  - Added support for DPoP (Demonstrating Proof-of-Possession) tokens validation. The feature is available by enabling `proof_of_possession_dpop`
  - Add support for JWT Secured Authorization Requests (JAR) on Authorization and Pushed Authorization (PAR) endpoints, see: `config.require_signed_request_object`
  - Add support for JARM response modes: `query.jwt`, `form_post.jwt`, `fragment.jwt`, `jwt`
- prometheus:
  - Added workspace label to Prometheus plugin metrics.
- AI Proxy now reads most prompt tuning parameters from the client, while the plugin config parameters under `model_options` are now just defaults. This fixes support for using the respective provider's native SDK.
- AI Proxy now has a `preserve` option for `route_type`, where the requests and responses are passed directly to the upstream LLM. This is to enable compatibility with any and all models and SDKs that may be used when calling the AI services.
- Addded support for EdDSA algorithms in JWT plugin
- Added support for ES512, PS256, PS384, PS512 algorithms in JWT plugin

allow to implement a rate limit by AI provider.

using the Azure provider with AI Proxy.

for JWT-Signer. We can achieve the goal of passing multiple values to a key by passing a JSON string as
_token_claims, add_channel_token_claims, set_channel_token_claims for individually adding claims to
ve_access_token_claims and remove_channel_token_claims to support the removal of claims.

C

penSSL 3.x.

meter, which enables Wasm filters to use the Proxy-Wasm operations `get_shared_data` and

eprecated fields

abling individual filters

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Added the `latencies.receive` property to the log serializer

Admin API

- Add LHS brackets filtering to search fields
- **Audit Log:** Add request_timestamp to audit_objects.
- **Audit Log**: Add before / after aliases for LHS Brackets filters.
- **Audit Log:** Allow audit_requests and audit_objects to be filtered by request_timestamp.
- **Audit Log:** change default ordering of audit_requests to sorted by request_timestamp descending

Bugfix

Plugin

- acme:
  - Fixed an issue where the certificate was not successfully renewed during ACME renewal.
  - Fixed migration of redis configuration.
  - fix a bug where the wrong error log is printed, regarding private keys.
- ai-proxy:
  - Fixed the bug that the `route_type` `/llm/v1/chat` didn't include the analytics in the responses.
- aws-lambda:
  - Fixed an issue where the latency attributed to AWS Lambda API requests was counted as part of the latency in Kong.
- degraphql:
  - Fixed an issue where GraphQL variables were not being correctly parsed and coerced into their defined types.
- jwt:
  - Fixed an issue where the plugin would fail when using invalid public keys for ES384 and ES512 algorithms.
- ldap-auth-advanced:
  - fix an issue where if the credential is encoded with no username kong will throw an error and return 500
  - fix an issue where an exception will be thrown when ldap search fails
- opentelemetry:
  - Fixed an OTEL sampling mode Lua panic bug, which happened when the `http_response_header_for_traceid` option was enabled.
- rate-limiting-advanced:
  - Refactored `kong/tools/public/rate-limiting` to keep the original interfaces unchanged (backward compatibility) and extend a new interface `new_instance` to provide isolation between different plugins. If you are using custom Rate Limiting plugins based on this library, please update the initialization code to the new format like 'local ratelimiting = require("kong.tools.public.rate-limiting").new_instance("custom-plugin-name")'. The old interface will be removed in the upcoming major release.
  - Fixed an issue where RLA and other similar plugins using the `rate-limiting` library, when used together, would interfere with each other and thus fail to

redis goes down

n being executed instead of being created to reduce some meaningless error logs

namespace have different configurations

value greater than `0` to `0`, the namespace will be cleared unexpectedly

cing timer can't be created or destroyed properly

n the Key Auth plugin.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Degraphql plugin now uses new configure handler to update graphql router with better error handling
- oas-validation, **WebSocket Size Limit**, **WebSocket Validator**, **XML Threat Protection**: priorities have been updated to prevent collisions between plugins. The relative priority (and the order of execution) of bundled plugins remained unchanged.

Core

- Fixed a bug where, if the the ulimit setting (open files) was low, Kong would fail to start as the `lua-resty-timer-ng` exhausted the available `worker_connections`. Decreased the concurrency range of the `lua-resty-timer-ng` library from `[512, 2048]` to `[256, 1024]` to fix this bug.
- Fixed an issue where `POST /config?flatten_errors=1` could not return a proper response if the input included duplicate upstream targets.
- **DNS Client**: Ignore a non-positive values on resolv.conf for options timeout, and use a default value of 2 seconds instead.
- Updated the file permission of `kong.logrotate` to 644.
- Fixed a problem on hybrid mode DPs, where a certificate entity configured with a vault reference may not get refreshed on time.
- Fixed the missing router section for the output of the request-debugging.
- Fixed an issue in the internal caching logic where mutexes could get never unlocked.
- Fixed an issue where the router didn't work correctly when the route's configuration changed.
- Fixed an issue where SNI-based routing didn't work using `tls_passthrough` and the `traditional_compatible` router flavor.
- Fixed a bug that `X-Kong-Upstream-Status` didn't appear in the response headers even if it was set in the `headers` parameter in the `kong.conf` file when the response was hit and returned by the Proxy Cache plugin.
- Fixed vault initialization by postponing vault reference resolving on init_worker
- Fixed a bug that allowed vault secrets to refresh even when they had no TTL set.
- **Vault**: do not use incorrect (default) workspace identifier when retrieving vault entity by prefix
- **Core**: Fixed unexpected table nil panic in the balancer's stop_healthchecks function
- Use `-1` as the worker ID of privileged agent to avoid access issues.
- Fix an issue where external plugins using the protobuf-based protocol would fail to call the `kong.Service.SetUpstream` method with an error `bad argument #2 to 'encode' (table expected, got boolean)`.
- Reverted the hard-coded limitation of the `ngx.read_body()` API in OpenResty upstreams' new versions when downstream connections are in HTTP/2 or HTTP/3 stream modes.
- Each Kong cache instance now utilizes its own cluster event channel. This approach isolates cache invalidation events and reducing the generation of unnecessary worker events.
- Updated telemetry collection for AI Plugins to allow multiple plugins data to be set for the same request.
- Improved the user experience in Kong Manager by fixing various UI-related issues.
- **Core**: Disable analytics in stream module to avoid unnecessary error logs.
- Fix a problem that a new DP cannot resolve the license required Vault reference after the first configuration push.
- Fixed an issue where DP was unable to resolve license required Vault reference when loading an existing lmdb.
- Fixing an issue where users were not allowed to start Kong Gateway if `admin_gui_auth_conf.scope` is missing `"openid"` or `"offline_access"` when `admin_gui_auth` is set to `openid-connect`. Kong Gateway will now print warning logs only if `"openid"` is missing from `admin_gui_auth_conf.scope`.

P

ays return a number, which was caused by an incorrectly stored string value in `ngx.ctx.host_port`.

d no longer includes the response receive time, so it now has the same value as the `X-Kong-Proxy-`
led in the new `latencies.receive` metric, so if desired, the old value can be calculated as
affects payloads from all logging plugins that use the log serializer: `file-log`, `tcp-log`, `udp-log`,`http-
ts for the HTTP Log Plugin's log format.

C

n from 1000 to 10000 for the `upstream_keepalive_max_requests` option.

r Python) would fail to apply a change to the plugin config via the Admin API.

n by default.

suite is set to `old`.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Admin API

- **Admin API**: fixed an issue where calling the endpoint `POST /schemas/vaults/validate` was conflicting with the endpoint `/schemas/vaults/:name` which only has GET implemented, hence resulting in a 405.
- The `/<workspace>/admins` endpoint was used to return admins associated with a workspace based on their assigned RBAC roles. It has been fixed to return admins according to the workspace they belong to.

CLI Command

- Fixed an issue where the `pg_timeout` was overridden to `60s` even if `--db-timeout` was not explicitly passed in CLI arguments.
- Fixed a bug that caused the `kong` command line tool to ignore the `lua_ssl_trusted_certificate` configuration option.

Clustering

- Adjust clustering compatible check related to AWS Secrets Manager
- Adjust clustering compatible check related to HCV kube auth path
- Adjusted a clustering compatible check related to Hashicorp Vault Approle authentication.
- Fixed a problem where event_hooks were prematurely validated in hybrid mode. The fix delays the validation of event_hooks to the point where event_hooks are emitted.

Kong Manager

- Fixed an issue where the “Add Role” button was visible when authenticating with an IDP. It is now hidden when Kong Manager is set to authenticate with an IDP.
- Corrected the documentation link shown on the RBAC user form page.

Dependency

Core

- Added package `tzdata` to DEB Docker image for convenient timezone setting.
- Bumped atc-router from v1.6.0 to v1.6.2
- Bumped libexpat to 2.6.2
- Bumped lua-kong-nginx-module from 0.8.0 to 0.11.0
- Bumped lua-protobuf to 0.5.1
- Bumped lua-resty-acme to 0.13.0
- Bumped lua-resty-aws from 1.3.6 to 1.4.1
- Bumped lua-resty-http to 0.17.2.
- Bumped lua-resty-lmdb from 1.4.1 to 1.4.2
- Bumped lua-resty-openssl from 1.2.0 to 1.3.1
- Bumped lua-resty-timer-ng to 0.2.7
- Bumped LuaRocks from 3.9.2 to 3.11.0

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

9d1aa9128ec36b4526

0.43

xpected input.

cket#427.

- Updated libpasswdqc to 2.0.3
- Updated lua-resty-cookie to 0.2.0
- Updated lua-resty-passwdqc to 2.0
- Updated xmlua to 1.2.1
- Updated libxml2 to 2.12.6
- Updated libxslt to 1.1.39
- Updated msgpack-c to 6.0.1
- Remove lua-resty-openssl-aux-module dependency

Performance

Plugin

- opentelemetry:
  - Increased queue max batch size to 200.

Performance

- Improved proxy performance by refactoring internal hooking mechanism.
- Sped up the router matching when the `router_flavor` is `traditional_compatible` or `expressions`.
- Speeded up tracing mechanism.

Known Issues

Core

- The Brotli module is missing from all of the following ARM64 Kong Gateway Docker images:
  - RHEL 9
  - Debian 12
  - Amazon Linux 2
  - Amazon Linux 2023

3.6.1.8

Release date 2024/10/11

Feature

Plugin

- aws-lambda:
  - Added support for a configurable STS endpoint with the new configuration field `aws_sts_endpoint_url`.

Core

I Database Authentication, with new configuration fields: “pg_iam_auth_assume_role_arn”, sume_role_arn”, and “pg_ro_iam_auth_role_session_name.”

IAM Authentication, with new configuration fields: `pg_iam_auth_sts_endpoint_url` and

Vault. This can either be configured by `vault_aws_sts_endpoint_url` as a global configuration, or

erwritten during upgrade. When upgrading, set the environment variable o avoid any interactive prompts and enable fully automatic upgrades.

on-default workspace will not take effect.

during `resurrect_ttl` time and could not be fetched by other workers.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Moved internal Unix sockets to a subdirectory (`sockets`) of the Kong prefix.
- Shortened names of internal Unix sockets to avoid exceeding the socket name limit.
- Fixed an issue where luarocks-admin was not available in /usr/local/bin.

Plugin

- ldap-auth-advanced:
  - fix an issue where an exception will be thrown when ldap search fails
- opentelemetry:
  - Fixed an issue where header_type being nil caused a concatenation error.
- rate-limiting-advanced Fixed an issue where if the `window_size` in the consumer group overriding config is different from the `window_size` in the default config, the rate limiting of that consumer group would fall back to local strategy.
- rate-limiting-advanced Fixed an issue where the sync timer may stop working due to race condition.

Dependency

Core

- Bumped lua-resty-aws to 1.5.3 to fix a bug related to STS regional endpoint.
- Made the RPM package relocatable with the default prefix set to `/`.

3.6.1.7

Release date 2024/07/09

Deprecation

Core

- Debian 10, CentOS 7, and RHEL 7 reached their End of Life (EOL) dates on June 30, 2024. As of this patch, Kong is not building Kong Gateway 3.6.x installation packages or Docker images for these operating systems. Kong is no longer providing official support for any Kong version running on these systems.

Feature

Plugin

- aws-lambda:
  - A new configuration field `empty_arrays_mode` is now added to control whether Kong should send `[]` empty arrays (returned by Lambda function) as `[]` empty arrays or `{}` empty objects in JSON responses.`

Dependency

Core

- Bumped lua-resty-events to 0.3.0

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ODITIONAL SECTION in DNS responses.

Admin API

- Add LHS brackets filtering to search fields
- **Audit Log:** Add request_timestamp to audit_objects.
- **Audit Log**: Add before / after aliases for LHS Brackets filters.
- **Audit Log:** Allow audit_requests and audit_objects to be filtered by request_timestamp.
- **Audit Log:** change default ordering of audit_requests to sorted by request_timestamp descending

Bugfix

CLI Command

- Fixed an issue where the `pg_timeout` was overridden to `60s` even if `--db-timeout` was not explicitly passed in CLI arguments.

Plugin

- acme:
  - Fixed migration of redis configuration.
- basic-auth:
  - Fix an issue of realm field not recognized for older kong versions (before 3.6)
- openid-connect:
  - Fixed a bug where anonymous consumers may be cached as nil under a certain condition.
- rate-limiting:
  - Fixed migration of redis configuration.
- request-validator:
  - Fix an issue where the plugin may fail to handle requests when param_schema is $ref schema.
  - Added a new configuration field `content_type_parameter_validation` to determine whether to enable Content-Type parameters validation.
- response-ratelimiting:
  - Fixed migration of redis configuration.

Core

- fix a bug that the `host_header` attribute of upstream entity can not be set correctly in requests to upstream as Host header when retries to upstream happen.
- Built-in RBAC roles for admins (`admin` under the default workspace and `workspace-admin` under non-default workspaces) now disallow CRUD actions to `/groups` and `/groups/*` endpoints.
- **Vitals**: Fixed a bug that each data plane connecting to the control plane would trigger the control plane to create a redundant table rotater timer.

Admin API

- The `/<workspace>/admins` endpoint was used to return admins associated with a workspace based on their assigned RBAC roles. It has been fixed to return admins

K

vas unavailable because the official documentation was removed after the 3.4.x.

I

C

e error logging.

ce active healthcheck timer usage.

ected input.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- rate-limiting-advanced Improved that timer spikes do not occur when there is network instability with the central data store.

3.6.1.4

Release date 2024/05/14

Feature

Plugin

- mtls-auth:
  - Add `default_consumer` option that allows a default consumer to be used when the client certificate is valid but does not match any existing consumers.

Bugfix

PDK

- **PDK:** fix kong.request.get_forwarded_port to always return a number which was caused by an incorrectly stored string value in ngx.ctx.host_port.

Core

- Fixed a problem that in hybrid DP mode a certificate entity configured with vault reference may not get refreshed on time
- fix vault initialization by postponing vault reference resolving on init_worker

Clustering

- Fixed a problem where event_hooks were prematurely validated in hybrid mode. The fix delays the validation of event_hooks to the point where event_hooks are emitted.

Plugin

- rate-limiting-advanced:
  - Refactored `kong/tools/public/rate-limiting` to keep the original interfaces unchanged (backward compatibility) and extend a new interface `new_instance` to provide isolation between different plugins. If you are using custom Rate Limiting plugins based on this library, please update the initialization code to the new format like 'local ratelimiting = require("kong.tools.public.rate-limiting").new_instance("custom-plugin-name")'. The old interface will be removed in the upcoming major release.
- oas-validation, **WebSocket Size Limit**, **WebSocket Validator**, **XML Threat Protection**: priorities have been updated to prevent collisions between plugins. The relative priority (and the order of execution) of bundled plugins remained unchanged.

Dependency

Core

- Bump lua-protobuf to 0.5.1

3.6.1.3

R

F

P

K

g 404 error if the `admin_gui_path` was not a slash.

3

R

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Feature

Plugin

- oas-validation:
  - Add a new field `api_spec_encoded` to indicate whether the `api_spec` is URI-Encoded.

Bugfix

Plugin

- acme:
  - Fixed an issue where the certificate was not successfully renewed during ACME renewal.
- degraphql:
  - Fixed an issue where GraphQL variables were not being correctly parsed and coerced into their defined types.
- rate-limiting-advanced:
  - Fixed an issue where RLA and other similar plugins using the `rate-limiting` library, when used together, would interfere with each other and thus fail to synchronize counter data to the central data store

Configuration

- Fix an issue where an external plugin (Go, Javascript, or Python) would fail to apply a change to the plugin config via the Admin API.

Core

- update file permission of kong.logrotate to 644
- **Vault**: do not use incorrect (default) workspace identifier when retrieving vault entity by prefix
- Fix a problem that a new DP cannot resolve the license required Vault reference after the first configuration push.
- Fixing an issue where users were not allowed to start Kong Gateway if `admin_gui_auth_conf.scope` is missing `"openid"` or `"offline_access"` when `admin_gui_auth` is set to `openid-connect`. Kong Gateway will now print warning logs only if `"openid"` is missing from `admin_gui_auth_conf.scope`.

Clustering

- Adjust clustering compatible check related to AWS Secrets Manager

Kong Manager

- fixed the display of the remaining days of license expireation date
- Fix an issue that setting up Developer Portal configuration `Developer Meta Fields` contains characters outside of the Latin1 range, admins are not able to login to Kong Manager.
- change the type of rbac token for the RBAC user to `password`

Dependency

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

0.43

et#427.

3

R

E

C

- Fix the missing router section for the output of the request-debugging
- revert the hard-coded limitation of the ngx.read_body() API in OpenResty upstreams' new versions when downstream connections are in HTTP/2 or HTTP/3 stream modes.

Clustering

- Adjusted a clustering compatible check related to Hashicorp Vault Approle authentication.

Plugin

- rate-limiting-advanced Falling back to local strategy if sync_rate = 0 when redis goes down
- rate-limiting-advanced The plugin now creates counter syncing timers when being executed instead of being created to reduce some meaningless error logs
- rate-limiting-advanced fix an issue where if `sync_rate` is changed from a value greater than `0` to `0`, the namespace will be cleared unexpectedly
- rate-limiting-advanced fix some timer-related issues where the counter syncing timer can't be created or destroyed properly

Kong Manager

- Fix an issue where custom plugins were missing from the plugin select page.
- Fix an issue where the service was not prefilled in the route form while using Expressions router.

3.6.1.0

Release date 2024/02/26

Feature

Configuration

- now TLSv1.1 and lower is by default disabled in OpenSSL 3.x

Bugfix

Core

- Fix a bug where the ulimit setting (open files) is low Kong will fail to start as the lua-resty-timer-ng exhausts the available worker_connections. Decrease the concurrency range of the lua-resty-timer-ng library from [512, 2048] to [256, 1024] to fix this bug.

Plugin

- ldap-auth-advanced:
  - fix an issue where if the credential is encoded with no username kong will throw an error and return 500
- opentelemetry:
  - fix otel sampling mode lua panic bug when http_response_header_for_traceid option enable

Configuration

ite is set to old

auth path

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- azure-functions:
  - azure-functions plugin now eliminates upstream/request URI and only use `routeprefix` configuration field to construct request path when requesting Azure API
- oas-validation:
  - bypass schema validation when content-type is not `application/json`.
- saml:
  - adjust the priority of the SAML plugin to 1010 to correct the integration between the SAML plugin and other consumer-based plugins

Core

- **BREAKING:** To avoid ambiguity with other Wasm-related nginx.conf directives, the prefix for Wasm `shm_kv` nginx.conf directives was changed from `nginx_wasm_shm_` to `nginx_wasm_shm_kv_`
- In OpenSSL 3.2, the default SSL/TLS security level has been changed from 1 to 2. Which means security level set to 112 bits of security. As a result RSA, DSA and DH keys shorter than 2048 bits and ECC keys shorter than 224 bits are prohibited. In addition to the level 1 exclusions any cipher suite using RC4 is also prohibited. SSL version 3 is also not allowed. Compression is disabled.

Admin API

- The listing endpoints for consumer groups (`/consumer_groups`) and consumers (`/consumers`) now respond with paginated results. The JSON key for the list has been changed to `data` instead of `consumer_groups` or `consumers`.

Deprecation

Plugin

- acme:
  - Standardize redis configuration across plugins. The redis configuration right now follows common schema that is shared across other plugins.
- rate-limiting:
  - Standardize redis configuration across plugins. The redis configuration right now follows common schema that is shared across other plugins.
- response-ratelimiting:
  - Standardize redis configuration across plugins. The redis configuration right now follows common schema that is shared across other plugins.

Feature

Plugin

- acl:
  - Add support for consumer-groups
- ip-restriction:
  - add support for consumer group scoping

umerated encoded with redundant leading padding

ut_redirect_uri can now be referenced as a secret in the Kong Vault

ion from headers

e (PKCE).

).

- rate-limiting-advanced:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - support to ratelimit by consumer group
- rate-limiting:
    - add support for consumer group scoping
    - support to ratelimit by consumer group
- request-termination:
    - add support for consumer group scoping
- Introduced the new ai-prompt-decorator plugin that enables prepending and appending llm/v1/chat messages onto consumer LLM requests, for prompt tuning.
- Introduced the new ai-prompt-guard which can allow and/or block LLM requests based on pattern matching.
- Introduced the new ai-prompt-template which can offer consumers and array of LLM prompt templates, with variable substitutions.
- Introduced the new ai-proxy plugin that enables simplified integration with various AI provider Large Language Models.
- Introduced the new ai-request-transformer plugin that enables passing mid-flight consumer requests to an LLM for transformation or sanitization.
- Introduced the new ai-response-transformer plugin that enables passing mid-flight upstream responses to an LLM for transformation or sanitization.
- Tracing Sampling Rate can now be set via the `config.sampling_rate` property of the OpenTelemetry plugin instead of it just being a global setting for the gateway.
- Add CONTROLLER_CERTIFICATE_FILE and CONTROLLER_CERTIFICATE_DIR env config for AppDynamics plugin to use self-signed certificate
- Support the `tls_client_auth` and `self_signed_tls_client_auth` auth methods in the OpenID Connect plugin, allowing to do mTLS Client Authentication with the IdP.
- rate-limiting-advanced Enhance the resolution of RLA sliding window weight

Core

- Adds telemetry collection for AI Proxy, AI Request Transformer, and AI Response Transformer, pertaining to model and provider usage.
- add ngx_brotli module to kong prebuild nginx
- Allow primary key passed as a full entity to DAO functions.
- Build deb packages for Debian 12. The debian variant of kong docker image is built using Debian 12 now.
- The expressions route now supports the `!` (not) operator, which allows creating routes like `!(http.path =^ "/a")` and `!(http.path == "/a" || http.path == "/b")`
- Add `source` property to log serializer, indicating the response is generated by `kong` or `upstream`.
- Ensure Kong-owned directories are cleaned up after an uninstall using the system's package manager.
- Support `http.path.segments.len` and `http.path.segments.*` fields in the expressions router which allows matching incoming (normalized) request path by individual segment or ranges of segments, plus checking the total number of segments.
- `net.src.*` and `net.dst.*` match fields are now accessible in HTTP routes defined using expressions.
- Extend support for getting and setting Gateway values via proxy-wasm properties in the `kong.*` namespace.
- add the `examples` field to the metaschema
- Add new `upstream_status` and `source` properties to analytics pusher.
- add the `consumer_groups` support for analytics

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

thentication method

up mapping feature (e.g., OIDC, LDAP) with Kong Manager, and also fix some issue with the group

all request ID fields, for better consistency

audit objects, and otherwise exclude singular key like `password` recursively.

A

anges. Also, introduces a new endpoint `/fips-status` to show its current status.

C

- **Clustering**: Expose data plane certificate expiry date on the control plane API.

- Resilience support for homogeneous Dataplane deployments. Now Dataplanes can act as importer and exporter at the same time, and Kong will try to control the concurrency when export the config.
- Data-plane nodes running in Konnect will now report config reload failures such as invalid configuration or transient errors to the control-plane.
- Print to log possible config options that causing DP to CP connection error.

Configuration

- display a warning message when Kong Manager is enabled but the Admin API is not enabled
- add DHE-RSA-CHACHA20-POLY1305 cipher to the intermediate configuration
- The default value of `dns_no_sync` option has been changed to `off`
- Allow to inject Nginx directives into Kong's proxy location block
- Validate LMDB cache by Kong's version (major + minor), wiping the content if tag mismatch to avoid compatibility issues during minor version upgrade.
- The default value of `dns_no_sync` option has been changed to `off`

PDK

- Increase the precision of JSON number encoding from 14 to 16 decimals

CLI Command

- Automatically reinitialize the workspace entity counters after executing the cli change migrations commands.

Kong Manager

- Added support for creating/editing the route-by-header plugin from the UI.
- Added an onboarding flow to make it easier for new customers to start using Kong Gateway.
- Now the summary section has a new design in both workspaces page and overview page.

Bugfix

Plugin

- datadog:
  - Fix a bug that datadog plugin is not triggered for serviceless routes. In this fix, datadog plugin is always triggered, and the value of tag `name`(service_name) is set as an empty value.
- forward-proxy:
  - Fixed the issue where request payload is being discarded when payload exceeded the `client_body_buffer_size`.
- jwt-signer:
  - support for consumer group scoping by using pdk `kong.client.authenticate` function
- ldap-auth-advanced:
  - fix some cache-related issues which cause groups_required to not work properly and unexpected return codes after a non-200 response

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - support for consumer group scoping by using pdk `kong.client.authenticate` function

- openid-connect:
    - Fix logout uri suffix detection by using normalized version of `kong.request.get_forwarded_path()` instead of `ngx.var.request_uri` (especially when passing query strings to logout)
    - remove unwanted argument `ignore_signature.userinfo` from the `userinfo_load` function
    - support for consumer group scoping by using pdk `kong.client.authenticate` function
    - fix the cache key collision when config `issuer` and `extra_jwks_uris` contain the same uri
    - Correctly handle boundary conditions for token expiration time checking
    - update time when calculating token expire
- proxy-cache-advanced:
    - remove undesired `proxy-cache-advanced/migrations/001_035_to_050.lua` that blocks migration from OSS to EE. This is a breaking change for customers using Kong Gateway between `0.3.5` and `0.5.0`.
- rate-limiting:
    - fix to provide better accuracy in counters when sync_rate is used with the redis policy.
    - fix an issuer where all counters are synced to the same DB at the same rate.
- saml:
    - support for consumer group scoping by using pdk `kong.client.authenticate` function
- Add missing WWW-Authenticate headers to 401 response in basic auth plugin.
- Enhance error responses for authentication failures in the Admin API
- forward-proxy fallback to the non-streaming proxy when the request body has already been read
- oas-validation Fix a bug where the plugin throws a runtime error caused by the ref parameter schema not being dereferenced.
- Expose metrics for serviceless routes
- Mark the `authorization_value` in the `oauth2-introspection` plugin as an encrypted field
- Mark the `introspection_headers_values` in the `openid-connect` plugin as an encrypted and referenceable field
- Fix typo in jwe-decrypt error message
- check if `sync_rate` is `nil` when calling the RLA phase `configure()`
- Skip sync with DB or Redis if `sync_rate` is `nil` or `null`.
- mtls-auth print notice log if revocation check fails with revocation_check_mode = IGNORE_CA_ERROR
- request-validator The Request Validator plugin now validates the request body schema when json is the suffix value in the request content type's subtype, for example `application/merge-patch+json`.
- rate-limiting-advanced Check the error of queries in the redis pipeline
- Provide better error messages in the route-transformer-advanced plugin
- validate private and public key for keys entity

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

entities and invalidate the related ca store caches when a ca cert is updated.

h allows more characters than the previous validation.

nitions. Improve performance as most schemas does not define transformations.

ingful response body when the internal error code 494 is triggered.

s router flavor are now case sensitive. This change does not affect on `traditional_compatible` mode the case.

riable to set ngx_wasm_module repository branch when building Kong.

hang risk

- **tracing:** Fixed an issue where a DNS query failure would cause a tracing failure.

- Expressions route in `http` and `stream` subsystem now have stricter validation. Previously they share the same validation schema which means admin can configure expressions route using fields like `http.path` even for stream routes. This is no longer allowed.
- **Tracing**: dns spans are now correctly generated for upstream dns queries (in addition to cosocket ones)
- **proxy-wasm**: Fixed "previous plan already attached" error thrown when a filter triggers re-entrancy of the access handler.
- Fixed an rbac issue that required adding missing endpoints to all workspaces.
- Dismiss confusing debug log from Redis tool of rate limiting #7077 #7101
- fix a bug where workload identity does not work for dataplane resilience
- Fix a bug that GCP backend vault hides the error message when secrets cannot be fetched
- fix the missing workspace_id in the output of request debugging when using the filter
- Eliminate asynchronous timer in syncQuery() to prevent hang risk
- Fixed critical level logs when starting external plugin servers. Those logs cannot be suppressed due to the limitation of OpenResty. We choose to remove the socket availibilty detection feature.
- Fix an issue where the IAM auth token was not refreshed when the underlying AWS credential expired.
- Print Redis's 'timeout' warning message only if it explicitly set. Use the default timeout value if it is not set.

Clustering

- Fix a bug causing data-plane status updates to fail when an empty PING frame is received from a data-plane
- Fix an issue where the dataplane's log serializer output has workspace name under Hybrid mode
- reduce message push error log when cluster_telemetry_endpoint config is disabled
- Return `-1` as worker id for privileged agent in the Clustering analytics.

Configuration

- fix error data loss caused by weakly typed of function in declarative_config_flattened function
- respect custom `proxy_access_log`

PDK

- response.set_header support header argument with table array of string
- Fix an issue that when using kong.response.exit, the Transfer-Encoding header set by user is not removed
- **Plugin Server**: fix an issue where every request causes a new plugin instance to be created

Admin API

- Fix an issue where the `/rbac/roles/:role/endpoints` endpoint did not accept `actions` as an array.
- The workspace listing API only shows workspaces that the current user has endpoints associated with
- Fix an issue where HTTP 500 errors were returned when paginating and sorting by timestamp fields (e.g., `created_at`).
- Fix an issue where unique violation errors were reported while trying to update the user_token with the same value on the same RBAC user.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

pts `roles` as arrays.

ting from CE to EE.

o prevent erroneous redirections to incorrect domains or protocols

Connect, including session, response_mode, and RP-initiated logout.

les from external sources (e.g., OIDC, LDAP).

specific keyset without permissions on the `/keys/*` endpoint.

- Fixed various issues while authenticating the Admin API via OpenID Connect.

Dependency

Core

- Fix incorrect LuaJIT LDP/STP fusion on ARM64 which may sometimes cause incorrect logic
- Bumped atc-router from 1.2.0 to 1.6.0
- Bumped kong-lapis from 1.14.0.3 to 1.16.0.1
- Bumped LPEG from 1.0.2 to 1.1.0
- Bumped lua-messagepack from 0.5.2 to 0.5.3
- Bumped lua-messagepack from 0.5.3 to 0.5.4
- Bumped lua-resty-aws from 1.3.5 to 1.3.6
- Bumped lua-resty-healthcheck from 3.0.0 to 3.0.1
- Bumped lua-resty-lmdb from 1.3.0 to 1.4.1
- Bumped lua-resty-timer-ng from 0.2.5 to 0.2.6
- Bump `ngx_wasm_module` to `a7087a37f0d423707366a694630f1e09f4c21728`
- Bumped OpenResty from 1.21.4.2 to 1.25.3.1
- Bumped OpenSSL from 3.1.4 to 3.2.1
- Bump resty-openssl from 0.8.25 to 1.2.0
- Bump `Wasmtime` version to `14.0.3`
- Bumped ngx_brotli to master branch, and disabled it on rhel7 rhel9-arm64 and amazonlinux-2023-arm64 due to toolchain issues
- Bumped lua-resty-healthcheck from 1.6.3 to 3.0.0
- bump submodule kong-openid-connect to 2.7.0
- Bump kong-redis-cluster to '1.5.2'
- Bump kong-redis-cluster to '1.5.3'
- bump jq to 1.7.1
- bump luasec to 1.3.2
- bump OpenSSL to 3.1.4
- bump lua-resty-aws to 1.3.6

Performance

Core

- Reuse match context between requests to avoid frequent memory allocation/deallocation

P

ibrary from [32, 256] to [512, 2048].

educe the impact to proxy path latency.

C

ord can be used for longer time in case of resolver downtime.

`ests` and `upstream_keepalive_max_requests` to `10000`. These changes are optimized to work better in
g, these new settings may have visible effects in loadbalancing - it can take more requests to start using all

ord can be used for longer time in case of resolver downtime.

P

and garbage-collections of spans

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Known Issues

Core

- The Brotli module is missing from all the following ARM64 Kong Gateway Docker images:
  - RHEL 9
  - Debian 12
  - Amazon Linux 2
  - Amazon Linux 2023

3.5.0.7

Release date 2024/07/09

Deprecation

Core

- Debian 10, CentOS 7, and RHEL 7 reached their End of Life (EOL) dates on June 30, 2024. As of this patch, Kong is not building Kong Gateway 3.5.x installation packages or Docker images for these operating systems. Kong is no longer providing official support for any Kong version running on these systems.

Feature

Plugin

- aws-lambda:
  - A new configuration field `empty_arrays_mode` is now added to control whether Kong should send `[]` empty arrays (returned by Lambda function) as `[]` empty arrays or `{}` empty objects in JSON responses.`

3.5.0.6

Release date 2024/06/22

Bugfix

Core

- Reverted DNS client to original behaviour of ignoring ADDITIONAL SECTION in DNS responses.

3.5.0.5

Release date 2024/06/18

Feature

Admin API

- Add LHS brackets filtering to search fields
- **Audit Log:** Add request_timestamp to audit_objects.

lters.

filtered by request_timestamp.

F

C

y can not be set correctly in requests to upstream as Host header when retries to upstream happen.

workspace and `workspace-admin` under non-default workspaces) now disallow CRUD actions to `/groups`

A

dmins associated with a workspace based on their assigned RBAC roles. It has been fixed to return admins

e current user has endpoints associated with

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- openid-connect:
  - Fixed a bug where anonymous consumers may be cached as nil under a certain condition.

Kong Manager

- Fixed an issue where the Dev Portal documentation link was unavailable because the official documentation was removed after the 3.4.x.

Dependency

Core

- Bumped lua-resty-azure from 1.4.1 to 1.5.0, to refine some error logging.
- Bumped lua-resty-events to 0.2.1
- Bumped lua-resty-healthcheck from 1.6.4 to 1.6.5, to reduce active healthcheck timer usage.

Performance

Plugin

- rate-limiting-advanced Improved that timer spikes do not occur when there is network instability with the central data store.

3.5.0.4

Release date 2024/05/20

Breaking Change

Core

- In OpenSSL 3.2, the default SSL/TLS security level has been changed from 1 to 2. Which means security level set to 112 bits of security. As a result RSA, DSA and DH keys shorter than 2048 bits and ECC keys shorter than 224 bits are prohibited. In addition to the level 1 exclusions any cipher suite using RC4 is also prohibited. SSL version 3 is also not allowed. Compression is disabled.

Feature

Admin API

- add gateway edition to the root endpoint of the admin api
- **Audit Log:** change default ordering of audit_requests to sorted by request_timestamp descending

Configuration

- now TLSv1.1 and lower is by default disabled in OpenSSL 3.x

Core

- which allows creating routes like `!(http.path =^ "/a")` and `!(http.path == "/a" || http.path ==`
- `ents.*` fields in the expressions router which allows matching incoming (normalized) request path by the total number of segments.
- le in HTTP routes defined using expressions.
- thentication method
- up mapping feature (e.g., OIDC, LDAP) with Kong Manager, and also fix some issue with the group

C

- nts. Now Dataplanes can act as importer and exporter at the same time, and Kong will try to control the

P

- ldap-auth-advanced:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- support decoding non-standard asn1 integer and enumerated encoded with redundant leading padding

- mtls-auth:
  - Add `default_consumer` option that allows a default consumer to be used when the client certificate is valid but does not match any existing consumers.

- oas-validation:
  - Add a new field `api_spec_encoded` to indicate whether the `api_spec` is URI-Encoded.

Bugfix

Core

- Header value matching (`http.headers.*`) in `expressions` router flavor are now case sensitive. This change does not affect on `traditional_compatible` mode where header value match are always performed ignoring the case.
- update file permission of kong.logrotate to 644
- Fixed a problem that in hybrid DP mode a certificate entity configured with vault reference may not get refreshed on time
- Fix the missing router section for the output of the request-debugging
- fix vault initialization by postponing vault reference resolving on init_worker
- **Vault**: do not use incorrect (default) workspace identifier when retrieving vault entity by prefix
- Expressions route in `http` and `stream` subsystem now have stricter validation. Previously they share the same validation schema which means admin can configure expressions route using fields like `http.path` even for stream routes. This is no longer allowed.
- Fixed an rbac issue that required adding missing endpoints to all workspaces.
- fix a bug where workload identity does not work for dataplane resilience
- Fix a problem that a new DP cannot resolve the license required Vault reference after the first configuration push.
- Fix a bug that GCP backend vault hides the error message when secrets cannot be fetched
- Fix an issue where external pluginservers would not start automatically with Kong

Plugin

- acme:
  - Fixed an issue where the certificate was not successfully renewed during ACME renewal.

- degraphql:
  - Fixed an issue where GraphQL variables were not being correctly parsed and coerced into their defined types.

- jwt-signer:
  - support for consumer group scoping by using pdk `kong.client.authenticate` function

- ldap-auth-advanced:
  - fix some cache-related issues which cause groups_required to not work properly and unexpected return codes after a non-200 response
  - fix an issue where if the credential is encoded with no username kong will throw an error and return 500
  - support for consumer group scoping by using pdk `kong.client.authenticate` function

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

eing validated.

`kong.client.authenticate` function

`kong.client.authenticate` function

sponse_header_for_traceid option enable

o keep the original interfaces unchanged (backward compatibility) and extend a new interface
nt plugins. If you are using custom Rate Limiting plugins based on this library, please update the

initialization code to the new format like ‘local ratelimiting = require(“kong.tools.public.rate-limiting”).new_instance(“custom-plugin-name”)’. The old interface will be removed in the upcoming major release.

  - Fixed an issue where RLA and other similar plugins using the `rate-limiting` library, when used together, would interfere with each other and thus fail to synchronize counter data to the central data store
  - Falling back to local strategy if sync_rate = 0 when redis goes down
  - The plugin now creates counter syncing timers when being executed instead of being created to reduce some meaningless error logs
  - fix an issue where if `sync_rate` is changed from a value greater than `0` to `0`, the namespace will be cleared unexpectedly
  - fix some timer-related issues where the counter syncing timer can’t be created or destroyed properly
- saml:
  - support for consumer group scoping by using pdk `kong.client.authenticate` function
- forward-proxy fallback to the non-streaming proxy when the request body has already been read
- Mark the `introspection_headers_values` in the `openid-connect` plugin as an encrypted and referenceable field
- oas-validation, **WebSocket Size Limit**, **WebSocket Validator**, **XML Threat Protection**: priorities have been updated to prevent collisions between plugins. The relative priority (and the order of execution) of bundled plugins remained unchanged.

PDK

- **PDK:** fix kong.request.get_forwarded_port to always return a number which was caused by an incorrectly stored string value in ngx.ctx.host_port.
- **OpenTelemetry**: Improved robustness of parsing for short trace IDs.

Configuration

- Fix an issue where an external plugin (Go, Javascript, or Python) would fail to apply a change to the plugin config via the Admin API.
- Set security level of gRPC’s TLS to 0 when ssl_cipher_suite is set to old

Clustering

- Adjust clustering compatible check related to AWS Secrets Manager
- Adjust clustering compatible check related to HCV kube auth path
- Fixed a problem where event_hooks were prematurely validated in hybrid mode. The fix delays the validation of event_hooks to the point where event_hooks are emitted.
- reduce message push error log when cluster_telemetry_endpoint config is disabled

Admin API

- Fix an issue where HTTP 500 errors were returned when paginating and sorting by timestamp fields (e.g., `created_at`).
- disallow admins or RBAC users to update their own roles

Portal

- Implement relative URLs for portal root path redirection to prevent erroneous redirections to incorrect domains or protocols

K

xpiration date was displayed inconsistently on the workspaces page and in the top tip.

uration “Developer Meta Field” to contain characters outside the Latin1 range prevented admin from

word

g 404 error if the `admin_gui_path` was not a slash.

I

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bumped OpenSSL from 3.1.4 to 3.2.0
- Bump resty-openssl from 0.8.25 to 1.2.0
- Improve the robustness of lua-cjson when handling unexpected input.
- Bump `kong-lua-resty-kafka` to `0.18`.
- Bumped lua-kong-nginx-module to 0.8.1
- Bump `lua-resty-luasocket` to `1.1.2` for fixing luasocket#427.
- bump lua-resty-healthcheck to 1.6.4
- bump lua-resty-aws to 1.3.6

Performance

Core

- Reuse match context between requests to avoid frequent memory allocation/deallocation

Configuration

- Bumped default values of `nginx_http_keepalive_requests` and `upstream_keepalive_max_requests` to `10000`.

Plugin

- opentelemetry:
  - increase queue max batch size to 200

3.5.0.3

Release date 2024/01/26

Breaking Change

Admin API

- The listing endpoints for consumer groups (`/consumer_groups`) and consumers (`/consumers`) now respond with paginated results. The JSON key for the list has been changed to `data` instead of `consumer_groups` or `consumers`.

Feature

Core

- Build deb packages for Debian 12. The debian variant of kong docker image is built using Debian 12 now.

Kong Manager

- added pagination support for nested consumer list and consumer group list.

Bugfix

K

s not show custom plugins.

non-default.

n't be created by `workspace-super-admin`.

D

C

3

R

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Breaking Change

Plugin

- saml:
  - adjust the priority of the SAML plugin to 1010 to correct the integration between the SAML plugin and other consumer-based plugins

Feature

Configuration

- The default value of `dns_no_sync` option has been changed to `off`

Plugin

- openid-connect:
  - extend `token_post_args_client` to support injection from headers
- openid-connect configurations `scopes`, `login_redirect_uri`, `logout_redirect_uri` can now be referenced as a secret in the Kong Vault

Bugfix

Core

- fix ldoc intermittent failure caused by LuaJIT error.
- Dismiss confusing debug log from Redis tool of rate limiting #7077 #7101
- fix the missing workspace_id in the output of request debugging when using the filter
- Eliminate asynchronous timer in syncQuery() to prevent hang risk
- Fixed critical level logs when starting external plugin servers. Those logs cannot be suppressed due to the limitation of OpenResty. We choose to remove the socket availibilty detection feature.

Plugin

- forward-proxy:
  - Fixed the issue where request payload is being discarded when payload exceeded the `client_body_buffer_size`.
- mocking:
  - Fix an issue where valid recursive schemas are always rejected.
- oas-validation:
  - Fixed an issue that the plugin throws a runtime error while validating parameters with AnyType schema and style keyword defined.
  - Fixed an issue that the nullable keyword did not take effect.
  - Fixed an issue that the URI component escaped characters were incorrectly unescaped.
- openid-connect:

version of `kong.request.get_forwarded_path()` instead of `ngx.var.request_uri` (especially when

me DB at the same rate.

e error caused by the ref parameter schema not being dereferenced.

ection plugin as an encrypted field

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fix an issue where the dataplane's log serializer output has workspace name under Hybrid mode

PDK

- **Plugin Server**: fix an issue where every request causes a new plugin instance to be created

Admin API

- Fix an issue where unique violation errors were reported while trying to update the user_token with the same value on the same RBAC user.

Dependency

Core

- Bumped OpenResty from 1.21.4.2 to 1.21.4.3
- Bump resty-openssl from 0.8.25 to 1.0.2
- bump luasec to 1.3.2

Performance

Configuration

- Bump `dns_stale_ttl` default to 1 hour so stale DNS record can be used for longer time in case of resolver downtime.

3.5.0.1

Release date 2023/11/14

Bugfix

Kong Manager

- Fix the issue where some values in config cards were not displayed correctly.

3.5.0.0

Release date 2023/11/08

Breaking Change

Plugin

- graphql-rate-limiting-advanced:
  - Fix a bug in the schema validation which prevents from using redis in cluster mode
- session:
  - a new configuration field `read_body_for_logout` was added with a default value of `false`, that changes behavior of `logout_post_arg` in a way that it is not anymore considered if the `read_body_for_logout` is not explicitly set to `true`. This is to avoid session plugin from reading request bodies by default on e.g. `POST` request for logout detection.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

were deprecated in Kong 3.4.

etwork header in crossing-origin pre-light requests

ndicate whether to include the base path when performing the path match evaluation.

- oas-validation:

  - Introduces a new property `include_base_path` to indicate whether to include the base path when performing the path match evaluation.

- openid-connect:
  - New field `unauthorized_destroy_session`, which when set to true, we destory the session (delete the user's session cookie) when the request is unauthorized. Default to true. Set to false to preserve the session.
  - New field `using_pseudo_issuer`. When set to true, the plugin instance will not discover configuration from the issuer.
- opentelemetry:
  - A new value is added to the parameter `header_type`, which allows Kong to inject datadog headers into the headers of requests forwarding to upstream.
- response-ratelimiting:
  - add support for secret rotation with redis connection
- add scan_count to redis storage schema
- 'openid-connect' plugin now supports public client
- Fix when the Dev portal OIDC is enabled, a 500 error is thrown when the administrator login successfully and then retrieves the session
- OpenID-Connect now support designate parameter name of token for introspection and revocation with `introspection_token_param_name` and `revocation_token_param_name` respectively.

Core

- Add a new endpoint `/schemas/vaults/:name` to retrieve the schema of a vault.
- rename `privileged_agent` to `dedicated_config_processing. Enable` dedicated_config_processing` by default
- Support observing the time consumed by some components in the given request.
- Plugins can now implement `Plugin:configure(configs)` function that is called whenever there is a change in plugin entities. An array of current plugin configurations is passed to the function, or `nil` in case there is no active configurations for the plugin.
- Add a request-aware table able to detect accesses from different requests.
- A unique Request ID is now populated in the error log, access log, error templates, log serializer, and in a new X-Kong-Request-Id header (configurable for upstream/downstream using the `headers` and `headers_upstream` configuration options).
- Add support for optional Wasm filter configuration schemas
- Support JSON in Wasm filter configuration
- aws vault backend use credential provider chain and support role assuming
- Allow OSS features to continue working with an expired license and configured Kong Enterprise features to continue operating in read-only mode. Kong Gateway now logs a daily critical message when a license is expired and within the 30 days grace period.
- Add troubleshooting tool
- add analytics_debug option to output requests to logs.
- **OpenID Connect:** Added support for mTLS proof of possession. The feature is available by enabling `proof_of_possession_mtls`

Clustering

- **Clustering**: Allow configuring DP metadata labels for on-premise CP Gateway

3 config backup `putObject` request

P

C

A

nt info, etc. to the report component. Also, add a checksum and timestamp to the output.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

PDK

- Fix several issues in Vault and refactor the Vault code base:
  - Make DAOs to fallback to empty string when resolving Vault references fail
  - Use node level mutex when rotation references
  - Refresh references on config changes
  - Update plugin referenced values only once per request
  - Pass only the valid config options to vault implementations
  - Resolve multi-value secrets only once when rotating them
  - Do not start vault secrets rotation timer on control planes
  - Re-enable negative caching
  - Reimplement the kong.vault.try function
  - Remove references from rotation in case their configuration has changed
- Fix response body gets repeated when `kong.response.get_raw_body()` is called multiple times in a request lifecycle.
- Tracing: fix an issue that resulted in some parent spans to end before their children due to different precision of their timestamps
- Fix a bug related to data interference between requests in the kong.log.serialize function.

Core

- Fix an issue that the TTL of the key-auth plugin didnt work in DB-less and Hybrid mode.
- Fix a problem that abnormal socket connection will be reused when querying Postgres database.
- Fix upstream ssl failure when plugins use response handler
- Fix an issue that protocol `tls_passthrough` can not work with expressions flavor
- fix the building failure when applying patches
- Vault references can be used in Dbless mode in declarative config
- Properly warmup Vault caches on init
- Vault resurrect time is respected in case a vault secret is deleted from a vault
- update vitals prometheus grafana dashboard
- Fix a keyring issue where a kong node fails to send keyring material when using cluster strategy
- Fix a bug that will cause a failure of sending tracing data to datadog when value of x-datadog-parent-id header in requests is a short dec string
- Enforce Content Security Policy (CSP) headers while serving static resources via Dev Portal and Kong Manager
- fix rbac retrieve group roles with the group name that type is number.
- Allow to specify azure_vault config options in the `kong.conf` file.
- Correctly invalidate caches based on names and ids for Consumer Groups
- Apply Nginx patch for detecting HTTP/2 stream reset attacks early (CVE-2023-44487)
- Hardcode part settings of `admin_gui_auth_conf` while `admin_gui_auth` is `openid-connect`

Plugin

- aws-lambda:

dential provider

-ascii characters.

cur in some scenarios when configuration notify_only_request_validation_failure is true

-ascii characters.

ays rejected.

code was revoking refresh token when it was supposed to revoke access token when using the discovered

t work.

- opentelemetry:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - fix an issue that resulted in invalid parent IDs in the propagated tracing headers
  - fix an issue that resulted in traces with invalid parent IDs when `balancer` instrumentation was enabled
- saml:
  - When the redis session storage is incorrectly configured, users now receive a 500 error instead of being redirected endlessly
  - Reduce severity of 'session was not found' messages to 'info'
- tcp-log:
  - fix an issue of unnecessary handshakes when reusing TLS connection
- Cache the AWS lambda service by those lambda service related fields
- mtls-auth should not cache the network failure when doing revocation check
- fix flooded json decoding warning logs
- allow the 'start' field be a past time
- Enhance error responses for authentication failures in the Admin API
- oas-validation Fix an issue where non `application/json` content-types are being rejected even though the request body is not required.
- Require a license to use Dynamic Plugin Ordering

Configuration

- The default value of `dns_no_sync` option has been changed to `on`
- Fix an issue that remove FIPS from free mode. Add a check of FIPS state and license type in validate_fips. If no license with FIPS `on` will throw an error and exit.
- lazily enable FIPS mode after a valid license is received and only emit a warning instead of blocking Kong to start.

Admin API

- Fix a bug that when an entity is deleted, the rbac_role_entities records of its cascaded entities are not deleted.
- Fix an issue that made it possible to create colliding routes in different workspaces when using `application/x-www-form-urlencoded` as content type in the admin API
- optimize the performance of querying plugins when access application_services/application_instances endpoint.
- Fix an issue where users cannot completely delete a developer by email via Admin API.

Portal

- sanitize developer name in portal email
- Support boolean token in portal email template and fix 500 error

Clustering

- Fix an issue where the dataplane hostname is `nil` in Vitals under Hybrid mode

Kong Manager

t in the certificate form do not accept multi-line content.

ssing on Teams page if no admin created on DB initialization with RBAC disabled

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

n ARM64

- Fix LDP/STP fusing for unaligned accesses on ARM64

- Bumped kong-lapis from 1.14.0.2 to 1.14.0.3
- Bump lua-kong-nginx-module from 0.6.0 to 0.8.0
- Bump Wasmtime version to 12.0.2
- Add troubleshooting tools to container images
- bump submodule kong-openid-connect to 2.5.7
- bump submodule kong-openid-connect to 2.5.9

CLI Command

- Update included curl to 8.4.0 & nghttp2 1.57.0

Plugin

- bump the dependency `kong-openid-connect` of oidc plugin from 2.5.5 to 2.5.7.

Performance

Core

- refactor workspace id and name retrieval
- workspaces.get_workspace() now tries to get workspace from cache instead of querying database directly

Configuration

- Bumped the default value of `upstream_keepalive_pool_size` to `512` and `upstream_keepalive_max_requests` to `1000`

Plugin

- rate-limiting-advanced to use the new Plugin:configure for building namespaces without looping through all the plugins

3.4.3.29

Release date 2026/08/10

Bugfix

Core

- Applied upstream nginx security patches for CVE-2026-42533.
- Applied upstream nginx security patches for CVE-2026-56434.
- Applied upstream nginx security patches for CVE-2026-60005.
- Applied upstream nginx security patches for CVE-2026-42055 and CVE-2026-48142.

Plugin

gin by always validating the `samlp:Response` signature when `idp_certificate` is configured correctly
ng a crafted unsigned SAML response from silently bypassing authentication under configurations that set

D

C

3

R

E

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Applied upstream nginx security patches for limiting the maximum number of headers (CVE-2026-49975).

3.4.3.27

Release date 2026/05/28

Bugfix

Core

- Applied upstream nginx security patches for CVE-2026-40701, CVE-2026-42934, CVE-2026-42946, CVE-2026-42945, and CVE-2026-9256.
- Security: Fixed CVE-2026-6338 in core proxy request header handling by preserving inbound `Content-Length` headers during hop-by-hop Connection header cleanup, preventing CL.0 request smuggling with crafted `Connection: Content-Length` headers on requests with a body.

Dependency

Core

- Bumped libexpat from 2.7.3 to 2.8.1 to address the issues: CVE-2026-25210, CVE-2026-24515, CVE-2026-32778, CVE-2026-32777, CVE-2026-32776, CVE-2026-41080, and CVE-2026-45186.
- Bumped OpenSSL from 3.5.5 to 3.5.6 to address the issues: CVE-2026-31790, CVE-2026-2673, CVE-2026-28387, CVE-2026-28388, CVE-2026-28389, CVE-2026-28390, and CVE-2026-31789.

3.4.3.26

Release date 2026/05/11

Bugfix

Core

- Fixed an issue where the real client IP from Proxy Protocol was not correctly reflected in access logs and logging plugins when nginx rejected requests early (497 plain HTTP to HTTPS port) or when the client disconnected prematurely during header parsing. Added an nginx patch to resolve `$remote_addr` from `$proxy_protocol_addr` on early request termination, and added `error_page` 497 routing through `kong_error_handler`.

Plugin

- request-validator:
  - Added `array_length_compat` config option to allow legacy Draft 4 schemas to use `minLength`/`maxLength` keywords for array length validation.

Kong Manager

- Fixed workspace list and routing not working correctly when workspace count exceeds 1000.

Dependency

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- graphql-rate-limiting-advanced:
  - Fixed an issue where the graphql-rate-limiting-advanced plugin's API threw 500 errors when handling HTTP PUT requests.
- openid-connect:
  - Fixed an issue where client-provided headers could be forwarded to upstream services when the corresponding JWT claim was missing from the token. The plugin now clears client headers before processing `upstream_headers_names`, ensuring that only verified JWT claim values are sent to upstream or downstream.

Kong Manager

- Fixed workspace navigation for workspace names containing non-ASCII characters.

Performance

Plugin

- openid-connect:
  - Removed issuer discovery from schema to improve performance upon plugin initialization or updating. The issuer discovery will only be triggerd by client requests.

3.4.3.24

Release date 2026/03/11

Bugfix

Core

- Apply Nginx patch for a man-in-the-middle upstream response injection (CVE-2026-1642)

Dependency

Core

- Bumped OpenSSL from 3.2.1 to 3.5.5 to address the issue: CVE-2025-15467.

3.4.3.23

Release date 2026/01/07

Bugfix

Plugin

- kafka-upstream:
  - Fixed issue where Kafka producer cached TLS certificates, causing failures when certificates were updated. Now, the plugin properly reloads updated certificates.
- opentelemetry:

expectedly on control planes of hybird mode.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

incorrect deprecation notice on active healthchecks headers.

Postgres versions below 12.

Bugfix

Configuration

- Fixed an issue where the restart command read configuration multiple times and failed to stop nginx.

Core

- Fixed an issue where `500` response status can occasionally occur during reconfiguration.

Plugin

- openid-connect:
  - Fixed an issue where the `client_credentials`/`authorization_code` auth would not auto-recover if IdP was not accessible during Kong startup.
- response-transformer-advanced:
  - Fixed an issue where the failure occurred when Nginx gzip was enabled for upstream.
- openid-connect Updated the rediscovery to use a short lifetime (5s) if the last discovery failed.

Clustering

- Fixed an issue where the data plane did not stop the health checker before cleaning up old configurations, which could lead to resource leaks.

Dependency

Core

- Bumped libexpat from 2.7.1 to 2.7.3.
- Bumped lua-resty-ljsonschema to 1.2.0. Fixed UTF-8 string length calculation and added support for `null` in `enum` types.

Performance

Core

- Improved the performance of retrieving workspace entity counts.

3.4.3.21

Release date 2025/09/05

Bugfix

Core

- Fixed a bug where the health checker could fail to initialize in rare cases.
- Fixed an issue where running Kong CLI commands with database configurations containing Hashicorp Vault references would fail to execute.
- **kafka**: Fixed an issue where the Kafka SCRAM authentication did not send the server nonce.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

y status API is not accurate on the control plane.

le values

esented in long form length couldn’t be decoded.

king plugin schema with object type. It should return an array type instead of an object type in the response

log messages in the error logs.

- Fixed an issue where the password field was required when editing basic-auth credentials, now allows empty passwords.
- Resolved a security issue in the Kong Manager UI.

Dependency

Core

- Bumped LuaRocks from 3.11.1 to 3.12.2.

3.4.3.20

Release date 2025/06/25

Bugfix

Plugin

- openid-connect:
  - Fixed an `500` error caused by receiving a JSON `null` from the request body when parsing bearer tokens or client IDs.

Dependency

Core

- Bumped luarocks from 3.11.1 to 3.12.1. This fixes the error `main function has more than 65536 constants`, which was preventing rocks from being published or installed.

3.4.3.19

Release date 2025/06/10

Bugfix

Configuration

- Fixed an issue where `debug_listen` incorrectly used the SSL-related configuration of `status_listen`.

Core

- Added a new configuration, `route_match_calculation`, which included bug fixes for the `traditional_compatible` router flavor. The default value `original` retained the current `3.x` router matching behavior without changes. The value `strict` enforced the router matching behavior with corrected calculation as follows:
  1. The weight (priority point) of a route was incorrectly downgraded.
  2. A route of HTTPS-only protocol with SNIs incorrectly captured requests over HTTP routes.
  3. A route of TLS-only protocol with SNIs incorrectly captured requests over TCP or UDP routes.

3.4.3.18

Release date 2025/04/29

F

P

) function for returning a non-normalized `forwarded_path`.

F

C

a certificate or CA certificate was configured with a Vault reference.

labels for a self-managed Gateway Control Plane.

P

redirect_uri errors when `config.redirect_uri` was not configured and the URI path contained spaces.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- session:
  - Fixed an issue where boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) stored the session's metadata in the database. This also resolves an issue with Dev Portal, where adding these fields to `portal_session_conf` wasn't working as expected.

Dependency

Core

- Bumped `lua-resty-openssl` from 1.2.1 to 1.5.1 to fix the worker process crash caused by a segmentation fault.

3.4.3.17

Release date 2025/03/26

Feature

Plugin

- session:
  - Added two boolean configuration fields `hash_subject` (default `false`) and `store_metadata` (default `false`) to store session's metadata in the database.

Core

- Added an option for GitHub Actions to build nginx/OpenResty with debug symbols.
- Added a new feature to invalidate the admin's or the developer's related session while changing the password.

Bugfix

Plugin

- app-dynamics:
  - Fixed segmentation fault caused by missing destructor call on process exit.
- ldap-auth-advanced:
  - Fixed an issue where binary string was truncated at the first null character.

Core

- **Vault**: Updated the AWS Vault supported regions list to the latest.

Dependency

Core

- Bumped libexpat from 2.6.2 to 2.6.4 to fix a crash in the XML_ResumeParser function caused by XML_StopParser stopping an uninitialized parser.
- Bumped lua-resty-events to 0.3.0
- Bumped lua-resty-healthcheck to 3.1.0

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue that certificate entity configured with vault reference may not get refreshed on time when initial with an invalid string.

Dependency

Core

- Bumped lua-kong-nginx-module from 0.8.1 to 0.8.2.

3.4.3.14

Release date 2024/12/17

Bugfix

Core

- Fixed an issue where the workspace id was not included in the plugin config in the plugins iterator.
- fix vault initialization by postponing vault reference resolving on init_worker
- Fixed an issue where using Hashicorp Vault AppRole authentication with a secret ID file would fail to read the secret ID.

Plugin

- graphql-rate-limiting-advanced:
  - Fixed an issue where the plugin may fail to authenticate to Redis correctly with vault-referenced redis configuration.
- mtls-auth:
  - Fixed an issue where a 500 error occurs when the configuration changes with the mTLS plugin enabled.
- rate-limiting-advanced:
  - Fixed an issue where counters of the overriding consumer groups didn't fetched when the window_size is different and the workspace is non-default.
  - Fixed an issue where multiple plugin instances sharing the same namespace enforced consumer groups and different `window_sizes` were used in the consumer group overriding configs, then the rate limiting of some consumer groups would fall back to local strategy. Now every plugin instance sharing the same namespace can set different `window_size`.
  - Fixed an issue where the plugin may fail to authenticate to Redis correctly with vault-referenced redis configuration.
  - Fixed an issue where RLA stores long expiration time items cause `no memory` errors.

3.4.3.13

Release date 2024/11/15

Feature

Plugin

- aws-lambda:
  - Added support for a configurable STS endpoint with the new configuration field `aws_sts_endpoint_url`.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ght

I Database Authentication, with new configuration fields: "pg_iam_auth_assume_role_arn", sume_role_arn", and "pg_ro_iam_auth_role_session_name."

IAM Authentication, with new configuration fields: `pg_iam_auth_sts_endpoint_url` and

Vault. This can either be configured by `vault_aws_sts_endpoint_url` as a global configuration, or

- Fixed an issue where the Vault secret cache got refreshed during `resurrect_ttl` time and could not be fetched by other workers.

- Moved internal Unix sockets to a subdirectory (`sockets`) of the Kong prefix.
- Shortened names of internal Unix sockets to avoid exceeding the socket name limit.
- Fixed an issue where luarocks-admin was not available in /usr/local/bin.
- Fixed a bug where analytics can break when the value type of rate-limiting related headers are not integer
- Fix an issue where the IAM auth token was not refreshed when the underlying AWS credential expired.

Plugin

- opentelemetry:
  - Fixed an issue where header_type being nil caused a concatenation error.
- rate-limiting-advanced:
  - Fixed an issue where the sync timer may stop working due to race condition.

Dependency

Core

- Bumped `LPEG` from `1.0.2` to `1.1.0` to keep the version consistent accross all active branches. The version bump includes fixes like UTF-8 ranges, Larger limit for rules and matches, accumulator capture, etc.
- Bumped lua-resty-aws to 1.5.3 to fix a bug related to STS regional endpoint.
- Bumped lua-resty-azure to 1.6.1 to fix a GET request build issue
- Made the RPM package relocatable with the default prefix set to `/`.

3.4.3.12

Release date 2024/08/08

Feature

Core

- Backported image signing and provenance generation to 3.4.

Bugfix

Core

- The kong.logrotate configuration file will no longer be overwritten during upgrade. This change will present an additional prompt for debian users upgrading via apt and .deb packages. To accept the defaults provided by Kong in the package use the following command `DEBIAN_FRONTEND=noninteractive apt upgrade kong-enterprise-edition_3.4.3.11_arm64.deb` depending on your architecture and the version upgrading to.
- Fixed a bug where a new data plane couldn't resolve a Vault reference after the first configuration push. This was happening due to issues with license pre-loading.

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Reverted DNS client to original behaviour of ignoring ADDITIONAL SECTION in DNS responses.

3.4.3.10

Release date 2024/06/18

Bugfix

Admin API

- The `/<workspace>/admins` endpoint was used to return admins associated with a workspace based on their assigned RBAC roles. It has been fixed to return admins according to the workspace they belong to.

Dependency

Core

- Bumped lua-resty-events to 0.2.1

3.4.3.9

Release date 2024/06/08

Feature

Admin API

- Add LHS brackets filtering to search fields
- **Audit Log:** Add request_timestamp to audit_objects.
- **Audit Log**: Add before / after aliases for LHS Brackets filters.
- **Audit Log:** Allow audit_requests and audit_objects to be filtered by request_timestamp.

Bugfix

Core

- Fixed an issue where cluster_cert or cluster_ca_cert is inserted into lua_ssl_trusted_certificate before being base64 decoded.
- fix a bug that the `host_header` attribute of upstream entity can not be set correctly in requests to upstream as Host header when retries to upstream happen.
- Built-in RBAC roles for admins (`admin` under the default workspace and `workspace-admin` under non-default workspaces) now disallow CRUD actions to `/groups` and `/groups/*` endpoints.
- **Vitals**: Fixed a bug that each data plane connecting to the control plane would trigger the control plane to create a redundant table rotater timer.

Admin API

- The workspace listing API only shows workspaces that the current user has endpoints associated with

P

tached as nil under a certain condition.

D

C

e error logging.

ce active healthcheck timer usage.

P

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- rate-limiting-advanced:
  - Improved that timer spikes do not occur when there is network instability with the central data store.

3.4.3.8

Release date 2024/05/16

Feature

Admin API

- **Audit Log:** change default ordering of audit_requests to sorted by request_timestamp descending

Bugfix

Admin API

- Fix an issue where HTTP 500 errors were returned when paginating and sorting by timestamp fields (e.g., `created_at`).

Plugin

- rate-limiting-advanced:
  - Refactored `kong/tools/public/rate-limiting` to keep the original interfaces unchanged (backward compatibility) and extend a new interface `new_instance` to provide isolation between different plugins. If you are using custom Rate Limiting plugins based on this library, please update the initialization code to the new format like 'local ratelimiting = require("kong.tools.public.rate-limiting").new_instance("custom-plugin-name")'. The old interface will be removed in the upcoming major release.
- oas-validation, **WebSocket Size Limit**, **WebSocket Validator**, **XML Threat Protection**: priorities have been updated to prevent collisions between plugins. The relative priority (and the order of execution) of bundled plugins remained unchanged.

Dependency

Core

- Improve the robustness of lua-cjson when handling unexpected input.
- Bumped `kong-lua-resty-kafka` to `0.19` for supporting TCP socket keepalive.

3.4.3.7

Release date 2024/04/23

Bugfix

PDK

- **PDK:** fix kong.request.get_forwarded_port to always return a number which was caused by an incorrectly stored string value in ngx.ctx.host_port.

Core

- Fixed a problem that in hybrid DP mode a certificate entity configured with vault reference may not get refreshed on time

C

nregistered event_hooks in hybrid mode

D

C

3

R

F

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- oas-validation:
  - Add a new field `api_spec_encoded` to indicate whether the `api_spec` is URI-Encoded.

Kong Manager

- Added support for Azure vault.

Bugfix

Plugin

- acme:
  - Fixed an issue where the certificate was not successfully renewed during ACME renewal.
- degraphql:
  - Fixed an issue where GraphQL variables were not being correctly parsed and coerced into their defined types.
- rate-limiting-advanced:
  - Fixed an issue where RLA and other similar plugins using the `rate-limiting` library, when used together, would interfere with each other and thus fail to synchronize counter data to the central data store

Configuration

- Fix an issue where an external plugin (Go, Javascript, or Python) would fail to apply a change to the plugin config via the Admin API.

PDK

- **OpenTelemetry**: Improved robustness of parsing for short trace IDs.

Kong Manager

- Fixed an issue where logging in fails when the Developer Portal configuration “Developer Meta Field” contains characters outside the Latin1 range.
- Fixed an issue where admin account profile page returns 404 error if `admin_gui_path` is not a slash.

Dependency

Core

- Bumped lua-kong-nginx-module to 0.8.1
- Bump `lua-resty-luasocket` to `1.1.2` for fixing luasocket#427.

3.4.3.5

Release date 2024/03/21

Breaking Change

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

…been changed from 1 to 2. Which means security level set to 112 bits of security. As a result RSA, DSA and …n 224 bits are prohibited. In addition to the level 1 exclusions any cipher suite using RC4 is also prohibited. …d.

…the schema of a vault.

…which allows creating routes like `!(http.path =^ "/a")` and `!(http.path == "/a" || http.path ==`

…nts in the given request.

…`)` function that is called whenever there is a change in plugin entities. An array of current plugin …re is no active configurations for the plugin.

…`ents.*` fields in the expressions router which allows matching incoming (normalized) request path by individual segment or ranges of segments, plus checking the total number of segments.

- `net.src.*` and `net.dst.*` match fields are now accessible in HTTP routes defined using expressions.
- aws vault backend use credential provider chain and support role assuming
- HashiCorp Vault backend now supports using Approle authentication method
- Allow OSS features to continue working with an expired license and configured Kong Enterprise features to continue operating in read-only mode. Kong Gateway now logs a daily critical message when a license is expired and within the 30 days grace period.
- Allow using RBAC token to authenticate while using group mapping feature (e.g., OIDC, LDAP) with Kong Manager, and also fix some issue with the group mapping feature.

Configuration

- now TLSv1.1 and lower is by default disabled in OpenSSL 3.x

PDK

- add support for Azure's KeyVault Secrets Engine.

Plugin

- cors:
  - Support the `Access-Control-Request-Private-Network` header in crossing-origin pre-light requests
- mtls-auth:
  - Add `default_consumer` option that allows a default consumer to be used when the client certificate is valid but does not match any existing consumers.

Bugfix

Core

- Header value matching (`http.headers.*`) in `expressions` router flavor are now case sensitive. This change does not affect on `traditional_compatible` mode where header value match are always performed ignoring the case.
- update file permission of kong.logrotate to 644
- **Vault**: do not use incorrect (default) workspace identifier when retrieving vault entity by prefix
- Expressions route in `http` and `stream` subsystem now have stricter validation. Previously they share the same validation schema which means admin can configure expressions route using fields like `http.path` even for stream routes. This is no longer allowed.
- Allow to specify azure_vault config options in the `kong.conf` file.

Plugin

- forward-proxy:
  - fallback to the non-streaming proxy when the request body has already been read
- jwt-signer:
  - support for consumer group scoping by using pdk `kong.client.authenticate` function
- ldap-auth-advanced:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ong.client.authenticate function

ong.client.authenticate function

ong.client.authenticate function

sponse_header_for_traceid option enable

redis goes down

n being executed instead of being created to reduce some meaningless error logs

value greater than `0` to `0`, the namespace will be cleared unexpectedly

- fix some timer-related issues where the counter syncing timer can't be created or destroyed properly

- saml:
  - support for consumer group scoping by using pdk `kong.client.authenticate` function
- Mark the `introspection_headers_values` in the `openid-connect` plugin as an encrypted and referenceable field

Configuration

- Set security level of gRPC's TLS to 0 when ssl_cipher_suite is set to old

Clustering

- reduce message push error log when cluster_telemetry_endpoint config is disabled

Admin API

- Fix a bug that when an entity is deleted, the rbac_role_entities records of its cascaded entities are not deleted.

Kong Manager

- fixed the display of the remaining days of license expireation date
- Conceal the user token input field while editing an RBAC user.

Dependency

Core

- Bumped atc-router from 1.2.0 to 1.6.0
- Bumped lua-resty-openssl to 1.2.1
- Bump `kong-lua-resty-kafka` to `0.18`.

Performance

Core

- Reuse match context between requests to avoid frequent memory allocation/deallocation

Configuration

- Bumped default values of `nginx_http_keepalive_requests` and `upstream_keepalive_max_requests` to `10000`.

Plugin

- opentelemetry:
  - increase queue max batch size to 200

3.4.3.4

R

F

C

nts. Now Dataplanes can act as importer and exporter at the same time, and Kong will try to control the

C

ined auth path when using HCV vault on Kubernetes.

F

C

- fix error data loss caused by weakly typed of function in declarative_config_flattened function

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- fix a bug where workload identity does not work for dataplane resilience
- Fix a bug that GCP backend vault hides the error message when secrets cannot be fetched
- Fix Ubuntu `18.04` manifest with OpenSSL `3.2.1` bump.
- Fix a bug that caused spans to not be instrumented with http.status_code when the request was not proxied to an upstream.

Plugin

- ldap-auth-advanced:
  - fix some cache-related issues which cause groups_required to not work properly and unexpected return codes after a non-200 response
  - fix an issue where if the credential is encoded with no username kong will throw an error and return 500

Dependency

Core

- Bumped OpenSSL from 3.1.4 to 3.2.1
- Bump resty-openssl from 0.8.25 to 1.2.0
- bump lua-resty-aws to 1.3.6

3.4.3.3

Release date 2024/01/17

Feature

Admin API

- add gateway edition to the root endpoint of the admin api

Core

- Build deb packages for Debian 12. The debian variant of kong docker image is built using Debian 12 now.

Plugin

- Add CONTROLLER_CERTIFICATE_FILE and CONTROLLER_CERTIFICATE_DIR env config for AppDynamics plugin to use self-signed certificate

Bugfix

Core

- Fixed an rbac issue that required adding missing endpoints to all workspaces.

Plugin

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

eing validated.

to prevent erroneous redirections to incorrect domains or protocols

how custom plugins.

reated by the role `workspace-super-admin`'s admin.

Dependency

Core

- Bump kong-redis-cluster to '1.5.3'
- bump lua-resty-healthcheck to 1.6.4

3.4.3.2

Release date 2023/12/22

Feature

Plugin

- ldap-auth-advanced:
  - support decoding non-standard asn1 integer and enumerated encoded with redundant leading padding

Bugfix

Admin API

- Optimize the performance of querying plugins when accessing the application_services/application_instances endpoints.

Kong Manager

- Fix an issue where clicking the spec upload input doesn't trigger file selection.
- Fix an issue where some services are missing from the Dev Portal's application list in Kong Manager.

3.4.3.1

Release date 2023/12/15

Breaking Change

Plugin

- saml:
  - adjust the priority of the SAML plugin to 1010 to correct the integration between the SAML plugin and other consumer-based plugins

Feature

Plugin

- openid-connect:
  - configurations `scopes`, `login_redirect_uri`, `logout_redirect_uri` can now be referenced as a secret in the Kong Vault
  - extend `token_post_args_client` to support injection from headers

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`a-resty-aws` as an underlying AWS library. The refactor simplifies the AWS-Lambda plugin code base and .

hrown when the administrator login successfully and then retrieves the session

cess log, error templates, log serializer, and in a new X-Kong-Request-Id header (configurable for `ostream` configuration options).

ed to `off`

Bugfix

Plugin

- aws-lambda:
  - let plugin-level proxy take effect on EKS IRSA credential provider
- forward-proxy:
  - Fixed the issue where request payload is being discarded when payload exceeded the `client_body_buffer_size`.
- mocking:
  - Fix the issue that path parameter cannot match non-ascii characters.
- oas-validation:
  - Fix a bug where the plugin throws a runtime error caused by the ref parameter schema not being dereferenced.
  - Fix an issue where valid recursive schemas are always rejected.
  - Fixed an issue that the plugin throws a runtime error while validating parameters with AnyType schema and style keyword defined.
  - Fixed an issue that the nullable keyword did not take effect.
  - Fixed an issue that the URI component escaped characters were incorrectly unescaped.
  - Fix the issue that path parameter cannot match non-ascii characters.
- openid-connect:
  - update time when calculating token expire
- rate-limiting:
  - fix an issuer where all counters are synced to the same DB at the same rate.
- tcp-log:
  - fix an issue of unnecessary handshakes when reusing TLS connection
- Cache the AWS lambda service by those lambda service related fields
- Mark the `authorization_value` in the `oauth2-introspection` plugin as an encrypted field
- Fix typo in jwe-decrypt error message

PDK

- Fix a bug related to data interference between requests in the kong.log.serialize function.
- **Plugin Server**: fix an issue where every request causes a new plugin instance to be created

Core

- print error message correctly when plugin fails
- fix ldoc intermittent failure caused by LuaJIT error.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

yieldable phases.

e config

onsumer Groups

nang risk

vers. Those logs cannot be suppressed due to the limitation of OpenResty. We choose to remove the socket

s workspace name under Hybrid mode

s under Hybrid mode

Admin API

- Fix an issue where unique violation errors were reported while trying to update the user_token with the same value on the same RBAC user.

Kong Manager

- Fix an issue where the Applications tab was not visible for services under non-default workspaces.
- Fix a known CVE in a dependency by upgrading it to a newer version.

Dependency

Core

- Bumped lua-resty-aws from 1.3.1 to 1.3.2
- Bumped OpenResty from 1.21.4.1 to 1.21.4.3
- Bump resty-openssl from 0.8.25 to 1.0.2
- Bump lua-resty-healthcheck from 1.6.2 to 1.6.3
- Bumped lua-resty-aws from 1.2.3 to 1.3.0
- Bumped lua-resty-aws from 1.3.2 to 1.3.5
- Bump lua-kong-nginx-module from 0.6.0 to 0.8.0
- bump jq to 1.7
- bump luasec to 1.3.2

CLI Command

- Backport troubleshooting tools to 3.4
- Update included curl to 8.4.0 & nghttp2 1.57.0

Performance

Configuration

- Bump `dns_stale_ttl` default to 1 hour so stale DNS record can be used for longer time in case of resolver downtime.

3.4.2.0

Release date 2023/11/10

Feature

Admin API

- Add counters such as routes, plugins, licenses, deployment info, etc. to the report component. Also, add a checksum and timestamp to the output.

B

C

acks early (CVE-2023-44487)

P

t work.

nt IDs when `balancer` instrumentation was enabled

C

ed to `on`

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Kong Manager

- Fix an issue where some services were not showing with the exact name or ID under certain circumstances while configuring a route.

Dependency

Core

- Bump ngx_wasm_module version to prerelease-0.1.1.
- Add troubleshooting tools to container images
- bump OpenSSL to 3.1.4

3.4.1.1

Release date 2023/10/12

Bugfix

Plugin

- saml:
  - Reduce severity of ‘session was not found’ messages to ‘info’

Core

- Apply Nginx patch for detecting HTTP/2 stream reset attacks early (CVE-2023-44487)

Dependency

Core

- bump libxml2 from 2.10.3 to 2.11.5

3.4.1.0

Release date 2023/09/28

Breaking Change

Plugin

- graphql-rate-limiting-advanced:
  - Fix a bug in the schema validation which prevents from using redis in cluster mode

Feature

Core

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

n when set to true, we destory the session (delete the user’s session cookie) when the request is
e the session.

, the plugin instance will not discover configuration from the issuer.

e, which allows Kong to inject datadog headers into the headers of requests forwarding to upstream.

  - Make DAOs to fallback to empty string when resolving Vault references fail
  - Use node level mutex when rotation references
  - Refresh references on config changes
  - Update plugin referenced values only once per request
  - Pass only the valid config options to vault implementations
  - Resolve multi-value secrets only once when rotating them
  - Do not start vault secrets rotation timer on control planes
  - Re-enable negative caching
  - Reimplement the kong.vault.try function
  - Remove references from rotation in case their configuration has changed
- Tracing: fix an issue that resulted in some parent spans to end before their children due to different precision of their timestamps
- Fix several issues in Vault and refactor the Vault code base:
  - Make DAOs to fallback to empty string when resolving Vault references fail
  - Use node level mutex when rotation references
  - Refresh references on config changes
  - Update plugin referenced values only once per request
  - Pass only the valid config options to vault implementations
  - Resolve multi-value secrets only once when rotating them
  - Do not start vault secrets rotation timer on control planes
  - Re-enable negative caching
  - Reimplement the kong.vault.try function
  - Remove references from rotation in case their configuration has changed
- Tracing: fix an issue that resulted in some parent spans to end before their children due to different precision of their timestamps

Plugin

- openid-connect:
  - Fix issue on token revocation on logout where the code was revoking refresh token when it was supposed to revoke access token when using the discovered revocation endpoint.
- opentelemetry:
  - fix an issue that resulted in invalid parent IDs in the propagated tracing headers
  - fix an issue that resulted in invalid parent IDs in the propagated tracing headers
- saml:
  - When the redis session storage is incorrectly configured, users now receive a 500 error instead of being redirected endlessly
- mtls-auth should not cache the network failure when doing revocation check
- allow the ‘start’ field be a past time

Core

- Fix a problem that abnormal socket connection will be reused when querying Postgres database.
- Fix upstream ssl failure when plugins use response handler
- Fix an issue that protocol `tls_passthrough` can not work with expressions flavor
- Fix an issue that the TTL of the key-auth plugin didnt work in DB-less and Hybrid mode.
- Fix a problem that abnormal socket connection will be reused when querying Postgres database.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

er

with expressions flavor

ng material when using cluster strategy

to datadog when value of x-datadog-parent-id header in requests is a short dec string

e is number.

vers. Those logs cannot be suppressed due to the limitation of OpenResty. We choose to remove the socket

D

C

may sometimes cause incorrect logic

may sometimes cause incorrect logic

3.4.0.0

Release date 2023/08/09

Feature

deployment

- Kong Gateway is now available on RHEL 9.

Core

- Introduced the `cascade` option for `/workspaces`, which lets you delete a workspace and all of its entities in one request.
- Consumer groups are now a core entity. With consumer groups, you can apply different configurations to select groups of consumers. The following plugins can now be scoped to consumer groups:
  - Rate Limiting Advanced
  - Request Transformer and Request Transformer Advanced
  - Response Transformer and Response Transformer Advanced

  See the documentation for consumer groups to learn more.
- Added a new `ttl` option to vault configurations, allowing users to define the interval at which references are automatically re-fetched from the configured vault.

  See the documentation for secrets rotation to learn more.
- The workspace name now appears in the logging payload.
- **Beta feature:** Introduced the beta of WebAssembly (`proxy-wasm`).#11218

  This release integrates `Kong/ngx-wasm-module`into Kong Gateway.
- The `/schemas` endpoint now returns additional information about cross-field validation as part of the schema. This should help tools that use the Admin API to perform better client-side validation.
- Enabled the `expressions` and `traditional_compatible` router flavors in the stream subsystem.#11071
- The upstream `host_header` and router `preserve_host` configuration parameters now work in stream TLS proxy.#11244
- In DB-less mode, the declarative schema is now fully initialized at startup instead of on-demand in the request path. This is most evident in decreased response latency when updating configuration via the `/config` API endpoint.#10932
- Tracing: Added the new attribute `http.route` to HTTP request spans.#10981
- Tracing: Added the span attribute `net.peer.name`, which records the upstream hostname if it's available in `balancer_data.hostname`. Thanks @backjo for contributing this change.#10723
- The default value of `lmdb_map_size` config has been bumped to `2048m`from `128m` to accommodate most commonly deployed config sizes in DB-less and hybrid modes.#11047
- The default value of `cluster_max_payload` config has been bumped to `16m`from `4m` to accommodate most commonly deployed config sizes in hybrid mode.#11090
- Removed Kong branding from the kong HTML error template.#11150

Plugin

- graphql-rate-limiting-advanced:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

stream targets.

rotocols.

245

elds_by_lua configuration for dynamic modification of log fields using Lua code.

his header can be turned off by setting `expose_error_code` to `false`.

token cache key by setting `token_cache_key_include_scope` to `true`.

header. The field `header_type`now accepts the `aws` value to handle this specific propagation

  - The `endpoint` parameter is now referenceable, and can be stored as a secret in a vault.#11220

- prometheus:
  - has been optimized to reduce proxy latency impacts during scraping. #10949#11040#11065
- rate-limiting-advanced:
  - The `redis` strategy now catches strategy connection failures.
- Validation for plugin queue related parameters has been improved.
- Validation for plugin queue related parameters has been improved. [#10840](https: `max_batch_size`, `max_entries`, and `max_bytes` are now declared as `integer` not `number`.
- Validation for plugin queue related parameters has been improved. [#10840](https: `initial_retry_delay` and `max_retry_delay` must now be numbers greater than 0.001 (in seconds).

Kong Manager

- Enhanced the user experience of editing pages for entities with a refined look and feel.
- Simplified the user path by removing the configuration pages for nested entities.

Bugfix

Core

- Fixed a potential memory leak and reconnection problem which could occur when telemetry breaks down due to any exceptions in its `send` thread.
- Telemetry: Fixed issues that broke the telemetry websocket:
  - Fixed an issue that caused the telemetry websocket to be blocked by latency while flushing Vitals to database. By using a queue as a buffer, the process of receiving Vitals data from the data plane is now decoupled from the process of flushing Vitals to the database on the control plane.
  - Fixed an issue that broke the telemetry websocket in Konnect mode due to unexpected payloads when the counter of requests equals zero.
- Fixed an issue where you could receive an empty `request_id` when generating audit data.
- Fixed an error that occurred when the header `x-datadog-parent-id` wasn't passed to Kong Gateway.
- Fixed a queueing-related issue that broke event hooks in 3.3.0.0.
- Updated the datafile library to make the SAML plugin work when Kong Gateway is controlled by systemd.
- Fixed an issue where a workspace couldn't attach to the cache's consumer well.
- Fixed a LuaJIT crash on Arm64 and enabled LuaJIT on M1.
- Fixed an issue where the license couldn't load when pulling `KONG_LICENSE_DATA` from a vault.
- Declarative config now performs proper uniqueness checks against its inputs. Previously, it would silently drop entries with conflicting primary/endpoint keys, or accept conflicting unique fields silently.#11199
- Fixed a bug where a worker consuming dynamic log level setting events used the wrong reference for notice logging.#10897
- Added a `User=` specification to the systemd unit definition so that Kong Gateway can be controlled by systemd again.#11066
- Fixed a bug that caused the sampling rate to be applied to individual spans, producing split traces.#11135
- Fixed a bug that caused the router to fail in `traditional_compatible` mode when a route with multiple paths and no service was created.#11158

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

rk correctly when `route.protocols` is set to `grpc` or `grpcs`.#11082

nfigure HTTPS redirection.#11166

ss mode by injecting the necessary directives into the Kong CLI `nginx.conf`.#11127#11291

would cause subsequent requests proxied through Kong Gateway to execute Go plugins with inconsistent same Go plugin is applied to different route or service entities.#11306

d

bling the Application Registration plugin didn't remove the **Register** button from a service.

Portal, where the UI would hang when attempting to expand an API.

A

rs=1 to throw an exception and return a 500 error under certain circumstances.#10896

- Fixed an issue where `/schemas/plugins/validate` endpoint failed to validate valid plugin configuration when the key of `custom_fields_by_lua` contained dot (`.`) character(s).#11091

status-api

- Removed the database information from the status API when operating in DB-less mode or on the data plane.#10995

Plugin

- **gRPC Gateway**:
  - Fixed an issue where an array with one element would fail to be encoded.
  - Fixed an issue where empty (all default value) messages couldn't be unframed correctly.#10836
- **mTLS Authentication**:
  - Fixed several revocation verification issues:
    - If `revocation_check_mode=IGNORE_CA_ERROR`, then the CRL revocation failure will be ignored.
    - Once a CRL is added into the store, it will always do CRL revocation check with this CRL file.
    - OCSP verification failed with `no issuer certificate in chain` error if the client only sent a leaf certificate.
    - `http_timeout` wasn't correctly set.
  - If `revocation_check_mode=IGNORE_CA_ERROR`, then the CRL revocation failure will be ignored.
  - Once a CRL is added into the store, it will always do CRL revocation check with this CRL file.
  - OCSP verification failed with `no issuer certificate in chain` error if the client only sent a leaf certificate.
  - `http_timeout` wasn't correctly set.
  - Optimized CRL revocation verification.
  - Fixed an issue that would cause an unexpected error when `skip_consumer_lookup` is enabled and `authenticated_group_by` is set to `null`.
- acme:
  - Fixed an issue where the sanity test didn't work with `kong` storage in hybrid mode.#10852
- mocking:
  - Fixed an issue where the plugin threw an error when the arbitrary elements were defined in the path node.
- oas-validation:
  - Fixed an issue where the plugin was unable to pass the validation even if path parameter was valid.
  - Fixed an issue where the plugin always validated the request body even if the method spec had no `requestBody` defined.
  - Fixed an issue where the comparison between large absolute value numbers could be incorrect due to the number being converted to exponential notation.
- oauth2-introspection:
  - Fixed an issue where the plugin failed when processing a request with JSON that is not a table.
- openid-connect:
  - Changed some log levels from `notice` to `error` for better visibility.
  - Correctly set the right table key on `log` and `message`.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

on fails, the plugin now prints the correct error.

`redis` policy. Thanks @giovanibrioni for contributing this change.#10559

ts.

d correctly when instrumentations were enabled.#10983

ed: Fixed an issue where the plugin wouldn't transform the response body when the upstream returned a

e plugin could lose connection to a broker when the broker leadership changed.

K

- Fixed an issue where Kong Manager didn't get the latest config when the Enterprise license was posted via the Admin API.
- Fixed incorrect CORS behavior that occurred when Kong Manager was integrated with the Portal GUI.
- Fixed an issue where OIDC in Kong Manager didn't handle `invalid credentials` when providing the wrong username.
- Added an alert message in the `admins tab` page for `workspace access` while the `admin_auth` is set to `openid-connect`.
- Fixed an issue where the custom permission endpoint didn't work for `/services/<service-name-or-id>/application_instances`.

Dependency

Core

- Bumped `kong-redis-cluster` from 1.5.0 to 1.5.1
- Bumped `lua-resty-ljsonschema` from 1.1.3 to 1.15
- Bumped `lua-resty-kafka` from 0.15 to 0.16
- Bumped `lua-resty-aws` from 1.2.2 to 1.2.3
- Bumped `lua-resty-openssl` from 0.8.20 to 0.8.23#10837#11099
- Bumped `kong-lapis` from 1.8.3.1 to 1.14.0.2#10841
- Bumped `lua-resty-events` from 0.1.4 to 0.2.0#10883#11083#11214
- Bumped `lua-resty-session` from 4.0.3 to 4.0.4#11011
- Bumped `OpenSSL` from 1.1.1t to 3.1.1#10180#11140
- Bumped `pgmoon` from 1.16.0 to 1.16.2 (Kong's fork)#11181#11229
- Bumped `atc-router` from 1.0.5 to 1.2.0#10100#11071
- Bumped `lua-resty-lmdb` from 1.1.0 to 1.3.0#11227

Breaking Change

Core

- **Cassandra DB support removed:** Cassandra DB support has been removed. It is no longer supported as a data store for Kong Gateway. #10931.
- **Alpine support removed:** Alpine packages and Docker images based on Alpine are no longer supported. Starting with Kong Gateway 3.4.0.0, Kong is not building new Alpine images or packages.#10926
- **Ubuntu 18.04 support removed** : Support for running Kong Gateway on Ubuntu 18.04 ("Bionic") is now deprecated, as Standard Support for Ubuntu 18.04 has ended as of June 2023. Starting with Kong Gateway 3.4.0.0, Kong is not building new Ubuntu 18.04 images or packages, and Kong will not test package installation on Ubuntu 18.04.

  If you need to install Kong Gateway on Ubuntu 18.04, see the documentation for previous versions.
- Amazon Linux 2022 artifacts are renamed to Amazon Linux 2023, based on AWS's own renaming.
- LMDB encryption has been disabled. The option `declarative_config_encryption_mode` has been removed from `kong.conf`.
- The `/consumer_groups/:id/overrides` endpoint is deprecated in favor of a more generic plugin scoping mechanism. See the new consumer groups entity.
- Renamed the configuration property `admin_api_uri` to `admin_gui_api_url`. The old `admin_api_uri` property is considered deprecated and will be fully removed

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

with the RHEL9 Docker image. The RHEL8 packages are still available from our package repository.

K

C

_endpoint field of the `http-log` plugin and the `endpoint` field of the `opentelemetry` plugin, do not

er manually or via the extension of `bundled` plugins), the `kong migrations finish` or `kong migrations` plugin schemas.

3

R

E

Plugin

- opentelemetry:
  - fix an issue that resulted in invalid parent IDs in the propagated tracing headers
- saml:
  - When the redis session storage is incorrectly configured, users now receive a 500 error instead of being redirected endlessly

Core

- Fix a problem that abnormal socket connection will be reused when querying Postgres database.
- Fix a keyring issue where a kong node fails to send keyring material when using cluster strategy
- Apply Nginx patch for detecting HTTP/2 stream reset attacks early (CVE-2023-44487)

PDK

- Tracing: fix an issue that resulted in some parent spans to end before their children due to different precision of their timestamps

Dependency

Core

- Fix incorrect LuaJIT LDP/STP fusion on ARM64 which may sometimes cause incorrect logic

3.3.1.0

Release date 2023/07/03

Deprecation

Core

- **Alpine deprecation reminder:** Kong has announced our intent to remove support for Alpine images and packages later this year. These images and packages are available in 3.2 and will continue to be available in 3.3. We will stop building Alpine images and packages in Kong Gateway 3.4.
- **Cassandra deprecation and removal reminder:** Using Cassandra as a backend database for Kong Gateway is deprecated. It is planned for removal with Kong Gateway 3.4.

Bugfix

Core

- Fixed a bug that causes `POST /config?flatten_errors=1` to throw an exception and return a 500 error under certain circumstances.
- Fixed a bug that would cause an error when the header `x-datadog-parent-id` is not passed to Kong.
- Fixed a queueing-related bug that meant the `event_hooks` did not fire and led to errors in the logs.
- Updated the datafile library that meant when Kong was started with systemd, the SAML plugin did not load.
- Fixed a bug that the anonymous report can't be silenced by setting `anonymous_reports=false`.

`0-alpine` was missing `resty.dns.resolver` patch.

orkspace with the cache's consumer well.

P

here a request with JSON that is not a table failed.:

3

R

F

C

de certificate of the backend Amazon S3 or GCP Cloud Storage service will now be validated if it goes

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- When managing secrets with an AWS or GCP backend, the backend server's certificate is now validated if it goes through HTTPS.
- Kong Enterprise now supports using AWS IAM database authentication to connect to the Amazon RDS (PostgreSQL) database.
- Kong Manager:
  - Kong Manager and Konnect now share the same UI for the navbar, sidebar, and all entity lists.
  - Improved display for the routes list when the expressions router is enabled.
  - **CA Certificates** and **TLS Verify** are now supported in the Kong Gateway service form.
  - Added a GitHub star in the free mode navbar.
  - Upgraded the Konnect CTA in free mode.
- SBOM files in SPDX and CycloneDX are now generated for Kong Gateway's Docker images.
- `runloop` and `init` error response content types are now compliant with the `Accept` header value.#10366
- You can now configure custom error templates.#10374
- The maximum number of request headers, response headers, URI arguments, and POST arguments that are parsed by default can now be configured with the following new configuration parameters:`lua_max_req_headers`, `lua_max_resp_headers`, `lua_max_uri_args`, and `lua_max_post_args`.#10443
- Added PostgreSQL triggers on the core entites and entities in bundled plugins to delete expired rows in an efficient and timely manner.#10389
- Added support for configurable node IDs.#10385
- Request and response buffering options are now enabled for incoming HTTP 2.0 requests.

  Thanks @PidgeyBE for contributing this change. #10204 #10595
- Added `KONG_UPSTREAM_DNS_TIME` to `ngx.ctx` to record the time it takes for DNS resolution when Kong proxies to an upstream.#10355
- Dynamic log levels now have a default timeout of 60 seconds.#10288

kong-gateway-with-konnect

- You can now configure labels for data planesto provide metadata information for Konnect.#10471
- Sending analytics to Konnect from Kong Gateway DB-less mode is now supported.

Admin API

- Added a new `updated_at` field for the following entities: `ca_certificates`, `certificates`, `consumers`, `targets`, `upstreams`, `plugins`, `workspaces`, `clustering_data_planes`, `consumer_group_consumers`, `consumer_group_plugins`, `consumer_groups`, `credentials`, `document_objects`, `event_hooks`, `files`, `group_rbac_roles`, `groups`, `keyring_meta`, `legacy_files`, `login_attempts`, `parameters`, `rbac_role_endpoints`, `rbac_role_entities`, `rbac_roles`, `rbac_users`, and `snis`.#10400
- The `/upstreams/<upstream>/health?balancer_health=1` endpoint always shows the balancer health through a new attribute: `balancer_health`. This always returns `HEALTHY` or `UNHEALTHY`, reporting the true state of the balancer, even if the overall upstream health status is `HEALTHCHECKS_OFF`. This is useful for debugging.#5885
- **Beta** : OpenAPI specs are now available for the Kong Gateway Admin API: Kong Gateway Admin API - Enterprise spec

status-api

- The `status_listen` server has been enhanced with the addition of the`/status/ready` API for monitoring Kong Gateway's health. This endpoint provides a `200` response upon receiving a `GET` request, but only if a valid, non-empty configuration is loaded and Kong Gateway is prepared to process user requests.

  Load balancers frequently utilize this functionality to ascertain Kong Gateway's availability to distribute incoming requests. #10610#10787
- **Beta** : An OpenAPI spec is now available for the Kong Gateway Status API.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

lugin.get_id.#9903

racing backends. See `kong.tracing` for details. #10577

ontent_type.

se to allow handling the cache key URI as lowercase.#10453

key in keys and key_sets.#9746

for Redis storage, which defaults to an empty string for backwards compatibility.#10562

- aws-lambda:
  - Added the configuration parameter `disable_https` to support scheme configuration on the lambda service API endpoint.#9799
- datadog:
  - The `host` configuration parameter is now referenceable.#10484
- http-log:
  - Added the `application/json; charset=utf-8` option for the `Content-Type` header to support log collectors that require that character set declaration.#10533
- jwt-signer:
  - Added the configuration field `add_claims`, which lets you add extra claims to JWT.
- opentelemetry:
  - Spans are now correctly correlated in downstream Datadog traces.10531
  - Added the `header_type` field. Previously, the `header_type` was hardcoded to `preserve`. Now it can be set to one of the following values: `preserve`, `ignore`, `b3`, `b3-single`,`w3c`, `jaeger`, or `ot`.#10620
  - Added the new span attribute `http.client_ip` to capture the client IP when behind a proxy.#10723
  - Added the `http_response_header_for_traceid` configuration parameter. Setting a string value in this field sets a corresponding header in the response.#10379
- proxy-cache:
  - Added the configuration parameter `ignore_uri_case` to allow handling the cache key URI as lowercase.#10453
- request-transformer-advanced:
  - The plugin now honors the following Kong Gateway configuration parameters: `untrusted_lua`, `untrusted_lua_sandbox_requires`, `untrusted_lua_sandbox_environment`. These parameters apply to advanced templates (Lua expressions).
- request-validator:
  - Errors are now logged for validation failures.
- zipkin and **OpenTelemetry**: These plugins now convert `traceid` in HTTP response headers to hex format.#10534

Bugfix

Core

- The Kong Enterprise systemd unit was incorrectly renamed to `kong.service` in 3.2.x.x versions. It has now been reverted back to `kong-enterprise-edition.service` to keep consistent with previous releases.
- Fixed an issue where Kong Gateway failed to generate a keyring when RBAC was enabled.
- Fixed `lua_ssl_verify_depth` in FIPS mode to match the same depth of normal mode.
- Removed the email field from the developer registration response.
- Websocket requests now generate balancer spans when tracing is enabled.
- Fixed an issue where management of licenses via the `/licenses/` endpoint would fail if the current license is not valid.
- [illegible]vould become mixed up when dynamic reordering was applied. This fix ensures proper sorting behavior in

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

[illegible]ong Manager would throw an error.
[illegible] became blank when selecting a tab that is currently active.
[illegible]ally jumped to `/login` instead.
[illegible]tsD plugin. This field appeared in Kong Manager under Metrics, but the field doesn't exist in the plugin's

[illegible]RC32 collision.#9856

[illegible]grade configuration for the AWS Lambda and Zipkin plugins for older versions of data planes.#10346
[illegible]e fields correctly for the Session plugin for older versions of data planes.#10352

[illegible]sionally skipped when the old-fashioned config style was used for DB-less Kong Gateway.#10348

[illegible]avior.#10364

- Fixed an issue where balancer passive healthchecks would use the wrong status code when Kong Gateway changed the status code from the upstream in the `header_filter` phase. #10325#10592
- Fixed an issue where schema validations failing in a nested record did not propagate the error correctly.#10449
- Fixed an issue where dangling Unix sockets would prevent Kong Gateway from restarting in Docker containers if it was not cleanly stopped.#10468
- Fixed an issue where the sorting function for traditional router sources or destinations led to `invalid order function for sorting` errors.#10514
- Fixed the UDP socket leak in `resty.dns.client` caused by frequent DNS queries.#10691
- Fixed a typo in the mlcache option `shm_set_tries`.#10712
- Fixed an issue where a slow startup of the Go plugin server caused a deadlock.#10561
- Tracing:
  - Fixed an issue that caused the `sampled` flag of incoming propagation headers to be handled incorrectly and only affect some spans.#10655
  - Fixed an issue that was preventing `http_client` spans from being created for OpenResty HTTP client requests.#10680
  - Fixed an approximation issue that resulted in reduced precision of the balancer span start and end times.#10681
  - `tracing_sampling_rate` now defaults to 0.01 (trace one of every 100 requests) instead of the previous 1 (trace all requests). Tracing all requests causes unnecessary resource drain for most production systems.#10774
- Fixed an issue with vault references, which caused Kong Gateway to error out when trying to stop.#10775
- Fixed an issue where vault configuration stayed sticky and cached even when configurations were changed.#10776
- Fixed the following PostgreSQL TTL clean-up timer issues:
  - Timers will now only run on traditional and control plane nodes that have enabled the Admin API.#10405
  - Kong Gateway now runs a batch delete loop on each TTL-enabled table with a number of `50.000` rows per batch.#10407
  - The cleanup job now runs every 5 minutes instead of every 60 seconds.#10389
  - Kong Gateway now deletes expired rows based on the database server-side timestamp to avoid potential problems caused by the differences in clock time between Kong Gateway and the database server.#10389
- Fixed an issue where an empty value for the URI argument `custom_id` crashed the `/consumer` API.#10475

kong-gateway-with-konnect

- The standard expired license notification no longer appears in logs for data planes running in Konnect mode (`konnect_mode=on`), as it does not apply to them.
- New license alert behavior for data planes running in Konnect mode:
  - If there are at least 16 days left before expiration, no alerts are issued.
  - If the license expires within 16 days, a warning level alert is issued every day.
  - If the license is expired, a critical level alert is issued every day.

PDK

- `request.get_uri_captures` now returns the unnamed part tagged as an array for jsonification.#10390
- Fixed an issue for tracing PDK where the sampling rate didn't work.#10485

Plugin

- **Forward Proxy**:
  - Fixed an issue which caused the wrong `latencies.proxy` to be used in the logging plugins. This plugin now evaluates `ctx.WAITING_TIME` in the forward proxy instead of doing it in the subsequent phase.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

stead of `statsd`.

ON payload caused an uncaught exception to be thrown during `pb.encode`.#10687

ncorrectly encoded as `"{}"`. They are now encoded as `"[]"` to comply with standards.#10790

horization, and returns a 403 HTTP code when a user isn't in the authorized groups.

mpty array when groups are not empty.

cached as `nil` if the wrong service was accessed first.#10522

ated by one plugin instance from being exchanged for an access token created by a different plugin

- opentelemetry:
  - Fixed an issue where reconfiguring the plugin didn't take effect.
  - Fixed an issue that caused spans to be propagated incorrectly resulting in the wrong hierarchy being rendered on tracing backends.#10663
- rate-limiting-advanced:
  - The `redis` rate limiting strategy now returns an error when Redis Cluster is down.
  - Fixed an issue where the rate limiting `cluster_events` broadcast the wrong data in traditional cluster mode.
  - The control plane no longer creates namespace or syncs.
- request-termination:
  - Fixed an issue with the `echo` option, which caused the plugin to not return the `uri-captures`.#10390
- request-transformer:
  - Fixed an issue where requests would intermittently be proxied with incorrect query parameters.10539
  - The plugin now honors the value of the `untrusted_lua` configuration parameter.#10327
- request-validator:
  - Fixed an issue where the validation function for the `allowed_content_types` parameter was too strict, making it impossible to use media types that contained a `-` character.
- jwe-decrypt, **OAS Validation**, and **Vault Authentication**: Added the missing schema field `protocols` for `jwe-decrypt`, `oas-validation`, and `vault-auth`.

Dependency

Core

- Updated the datafile library dependency to fix the following issues:
  - Kong Gateway didn't work when installed on a read-only file system.
  - Kong Gateway didn't work when started from systemd.
- Bumped `lua-resty-session` from 4.0.2 to 4.0.3#10338
- Bumped `lua-protobuf` from 0.3.3 to 0.5.0#10137#10790
- Bumped `lua-resty-timer-ng` from 0.2.3 to 0.2.5#10419#10664
- Bumped `lua-resty-openssl` from 0.8.17 to 0.8.20#10463#10476
- Bumped `lua-resty-http` from 0.17.0.beta.1 to 0.17.1#10547
- Bumped `lua-resty-aws` from 1.1.2 to 1.2.2
- Bumped `lua-resty-gcp` from 0.0.11 to 0.0.12
- Bumped `LuaSec` from 1.2.0 to 1.3.1#10528
- Bumped `lua-resty-acme` from 0.10.1 to 0.11.0#10562
- Bumped `lua-resty-events` from 0.1.3 to 0.1.4#10634
- Bumped `lua-kong-nginx-module` from 0.5.1 to 0.6.0#10288

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

intent to remove support for Alpine images and packages later this year. These images and packages are and packages in Kong Gateway 3.4.

Cassandra as a backend database for Kong Gateway is deprecated. It is planned for removal with Kong

ore compatible with the behavior of `traditional` mode by splitting routes with multiple paths into oduction of the new router in Kong Gateway 3.0, `traditional_compat` mode assigned only one priority to r expressions were mixed in a route. This was not how multiple paths were handled in the`traditional` eparate priority value is assigned to each path in a route.#10615

- **Tracing** : `tracing_sampling_rate` now defaults to 0.01 (trace one of every 100 requests) instead of the previous 1 (trace all requests). Tracing all requests causes unnecessary resource drain for most production systems.#10774

Plugin

- app-dynamics:
  - The plugin version has been updated to match Kong Gateway's version.
- http-log:
  - If the log server responds with a 3xx HTTP status code, the plugin now considers it to be an error and retries according to the retry configuration. Previously, 3xx status codes would be interpreted as a success, causing the log entries to be dropped.#10172
- zipkin:
  - This plugin now uses queues for internal buffering. The standard queue parameter set is available to control queuing behavior.#10753
- Plugin batch queuing: The module `kong.tools.batch_queue` has been renamed to `kong.tools.queue` and the API was changed. If your custom plugin uses queues, it must be updated to use the new API.#10172
- Plugin batch queuing: HTTP Log, StatsD, OpenTelemetry, and Datadog

  The queuing system has been reworked, causing some plugin parameters to not function as expected anymore. If you use queues in these plugins, new parameters must be configured. See each plugin's documentation for details.
- pre-function and **Post-function**: `kong.cache` now points to a cache instance that is dedicated to the Serverless Functions plugins. It does not provide access to the global Kong Gateway cache. Access to certain fields in `kong.conf` has also been restricted.#10417

Known Issues

Core

- Due to known issues, Kong recommends not enabling page-level LMDB encryption in versions 3.0.x-3.3.x.

  Don't set `declarative_config_encryption_mode`; leave it at its default value of `off`. Continue relying on disk-level encryption to encrypt the configuration on disk.
- When sending an invalid configuration to the `/config` endpoint while running in DB-less mode and with `flatten_errors=1` set, Kong Gateway incorrectly returns a 500. This should be a 400 because the configuration is invalid.
- When the OpenID Connect (OIDC) plugin is configured to reference HashiCorp Vault in the `config.client_secret` field (for example, `{vault://hcv/clientSecret}`), it does not look up the secret correctly.

3.2.2.5

Release date 2023/10/12

Bugfix

Core

- Applied Nginx patch for early detection of HTTP/2 stream reset attacks. This change is in direct response to the identified vulnerability CVE-2023-44487.
- Fixed a keyring issue where Kong Gateway nodes would fail to send keyring data when using the cluster strategy.
- Fixed an issue where an abnormal socket connection would be incorrectly reused when querying the PostgreSQL database.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

n, enabling Kong Gateway to be controlled by systemd again.#11066

ork failures when running certificate revocation checks.

endlessly redirected when the Redis session storage is incorrectly configured.

Release date 2023/09/15

Bugfix

Core

- Updated the datafile library to make the SAML plugin work again when Kong is controlled by systemd.
- Fixed an issue where the anonymous report couldn't be silenced by setting `anonymous_reports=false`.
- Fixed an issue where a crashing Go plugin server process would cause subsequent requests proxied through Kong to execute Go plugins with inconsistent configurations. The issue only affected scenarios where the same Go plugin is applied to different route or service entities.

Plugin

- openid-connect:
  - Correctly set the right table key on `log` and `message`.
  - If an invalid opaque token is provided but verification fails, print the correct error.
- rate-limiting-advanced:
  - The control plane no longer attempts to create namespace or synchronize counters with Redis.
- rate-limiting:
  - The redis rate limiting strategy now returns an error when Redis Cluster is down.
- response-transformer-advanced:
  - Does not load response body when `if_status` does not match.

Kong Manager

- Fixed an issue where the Zipkin plugin prevented users from editing the `static_tags` configuration.
- Fixed an issue where the unavailable Datadog Tracing plugin displayed on the plugin installation page.
- Fixed an issue where some metrics were missing from the StatsD plugin.
- Fixed an issue where locale files were not found when using a non-default `admin_gui_path` configuration.
- Fixed an issue where endpoint permissions for application instances did not work as expected.
- Fixed an issue where some icons were shown as unreadable symbols and characters.
- Fixed an issue where users were redirected to pages under the default workspace when clicking links for services or routes of entities residing in other workspaces.
- Fixed an issue that failed to redirect OpenID Connect in Kong Manager if it was provided with an incorrect username.

Dependency

Core

- `lua-resty-kafka` is bumped from 0.15 to 0.16
- Bumped `OpenSSL` from 1.1.1t to 3.1.1

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ong Gateway on Ubuntu 18.04 ("Bionic") is now deprecated, as Standard Support for Ubuntu 18.04 has
.4, Kong is not building new Ubuntu 18.04 images or packages, and Kong will not test package installation

ux 2023, based on AWS's own renaming.

are no longer supported in future versions.

Core

- **Alpine deprecation reminder:** Kong has announced our intent to remove support for Alpine images and packages later this year. These images and packages are available in 3.2 and will continue to be available in 3.3. We will stop building Alpine images and packages in Kong Gateway 3.4.

Bugfix

Core

- Fixed an error with the `/config` endpoint. If `flatten_errors=1` was set and an invalid config was sent to the endpoint, a 500 error was incorrectly returned.

3.2.2.2

Release date 2023/05/19

Bugfix

Core

- Fixed the OpenResty `ngx.print` chunk encoding duplicate free buffer issue that led to the corruption of chunk-encoded response data.#10816#10824
- Fixed the UDP socket leak in `resty.dns.client` caused by frequent DNS queries.#10691

Plugin

- rate-limiting-advanced:
  - Fixed the log flooding issue caused by low `sync_rate` settings.

3.2.2.1

Release date 2023/04/03

Deprecation

Core

- **Alpine deprecation reminder:** Kong has announced our intent to remove support for Alpine images and packages later this year. These images and packages are available in 3.2 and will continue to be available in 3.3. We will stop building Alpine images and packages in Kong Gateway 3.4.

Bugfix

Core

- Fixed the Dynatrace implementation. Due to a build system issue, Kong Gateway 3.2.x packages prior to 3.2.2.1 didn't contain the debug symbols that Dynatrace requires.

3.2.2.0

Release date 2023/03/22

Bugfix

C

artifacts incorrectly handled HTTP/2 requests, causing the protocol to fail. These artifacts have been

`-enterprise-edition`. This file was missing in all 3.x versions of Kong Gateway prior to this release.

P

tems.

`n_auth_ttl` (removed since 3.2.0.0).

ues with the Datadog Tracing plugin and elected to remove it from the 3.2 release. We plan to add the

Known issues

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Due to changes in GPG keys, using yum to install this release triggers a `Public key for kong-enterprise-edition-3.2.1.0.rhel7.amd64.rpm is not installed` error. The package *is* signed, however, it's signed with a different (rotated) key from the metadata service, which triggers the error in yum. To avoid this error, manually download the package from and install it.

3.2.1.0

Release date 2023/02/28

Deprecation

Core

- Deprecated Alpine Linux images and packages.

  Kong is announcing our intent to remove support for Alpine images and packages later this year. These images and packages are available in 3.2 and will continue to be available in 3.3. We will stop building Alpine images and packages in Kong Gateway 3.4.

Feature

Core

- Changed the underlying operating system (OS) for our convenience Docker tags (for example, `latest`, `3.2.1.0`, `3.2`) from Debian to Ubuntu.
- When `router_flavor` is set to`traditional_compatible`, Kong Gateway verifies routes created using the expression router instead of the traditional router to ensure created routes are compatible.#9987
- In DB-less mode, the `/config` API endpoint can now flatten all schema validation errors into a single array using the optional `flatten_errors` query parameter.#10161
- The upstream entity now has a new load balancing algorithm option: `latency`. This algorithm chooses a target based on the response latency of each target from prior requests.#9787
- The Nginx `charset` directive can now be configured with Nginx directive injections. Set it in Kong Gateway's configuration with `nginx_http_charset#10111`
- The services upstream TLS configuration is now extended to the stream subsystem.#9947
- Added the new configuration parameter `ssl_session_cache_size`, which lets you set the Nginx directive `ssl_session_cache`. This configuration parameter defaults to `10m`. Thanks Michael Kotten for contributing this change.#10021
- `status_listen` now supports HTTP2. #9919
- The shared Redis connector now supports username + password authentication for cluster connections, improving on the existing single-node connection support. This automatically applies to all plugins using the shared Redis configuration.
- **FIPS Support**:
  - The OpenID Connect, Key Authentication - Encrypted, and JWT Signer plugins are now FIPS 140-2 compliant. If you are migrating from Kong Gateway 3.1 to 3.2 in FIPS mode and are using the `key-auth-enc` plugin, you should send PATCH or POST requests to all existing `key-auth-enc` credentials to re-hash them in SHA256.
  - FIPS-compliant Kong Gateway packages now support PostgreSQL SSL connections.

Plugin

- **Improved Plugin Documentation**:

compatibility page.

ore clarity on supported network protocols and on entity scopes.

xamples:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - JWT Signer
    - Key Auth
    - OpenID Connect
    - Rate Limiting Advanced
    - SAML
    - StatsD
- **Plugin entity**:
    - Added an optional `instance_name` field, which identifies a particular plugin entity.#10077:
- aws-lambda:
    - Added the configuration parameter `aws_imds_protocol_version`, which lets you select the IMDS protocol version. This option defaults to `v1` and can be set to `v2` to enable IMDSv2.#9962
- canary:
    - Added a default value for the `start` field in the canary plugin. If not set, the start time defaults to the current timestamp.
- http-log:
    - The `headers` configuration parameter is now referenceable, which means it can be securely stored in a vault.#9948
- opa:
    - This plugin can now handle custom messages from the OPA server.
- opentelemetry:
    - This plugin can now be scoped to individual services, routes, and consumers.#10096
- statsd:
    - Added the `tag_style` configuration parameter, which allows the plugin to send metrics with tags. The parameter defaults to `nil`, which means that no tags are added to the metrics.#10118
- zipkin:
    - Added support for setting the durations of Kong phases as span tags through the configuration property `phase_duration_flavor`.#9891
- session, **OpenID Connect**, and **SAML**: These plugins now use `lua-resty-session` v4.0.0.

  This update includes new session functionalities such as configuring audiences to manage multiple sessions in a single cookie, global timeout, and persistent cookies.

  Due to this update, there are also a number of deprecated and removed parameters in these plugins. See the invidividual plugin documentation for the full list of changed parameters in each plugin.

    - Session changelog
    - OpenID Connect changelog
    - SAML changelog
- session, **OpenID Connect**, and **SAML**: Session changelog
- session, **OpenID Connect**, and **SAML**: OpenID Connect changelog
- session, **OpenID Connect**, and **SAML**: SAML changelog

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

nced: In hybrid and DB-less modes, these plugins now support `sync_rate = -1` with any strategy,

g `ttl`, so `ttl` is always returned as a whole integer.#9960

o the `Content-Type` response header when the upstream response doesn't contain it.#9905

on was loaded, the configuration hash was incorrectly set to the value

r grow without bounds if their consumers fail to process the entries. Instead, old batches are now dropped

- Fixed an issue where `X-Kong-Upstream-Status` couldn't be emitted when a response was buffered.#10056

- Improved the error message for invalid JWK entries.#9904
- Fixed an issue where the `#` character wasn't parsed correctly from environment variables and vault references.10132
- Fixed an issue where control plane didn't downgrade configuration for the AWS Lambda and Zipkin plugins for older versions of data planes.#10346
- Fixed an issue in DB-less mode, where validation of regex routes could be skipped when using a configuration format older than `3.0`.#10348
- Fixed an issue where the forward proxy between the data plane and the control plane didn't support telemetry port 8006.
- Fix the PostgreSQL mTLS error `bad client cert type`.
- Fixed issues with the Admin API's `/licenses` endpoint:
  - The Enterprise license wasn't being picked up by other nodes in a cluster.
  - Vitals routes weren't accessible.
  - Vitals wasn't showing up in hybrid mode.
- Fixed RBAC issues:
  - Fixed an issue where workspace admins couldn't add rate limiting policies to consumer groups.
  - Fixed an issue where workspace admins in one workspace would have admin rights in other workspaces. Workspace admins are now correctly restricted to their own workspaces.
  - Fixed a role precedence issue with RBAC. RBAC rules involving deny (negative) rules now correctly take precedence over allow (non-negative) roles.

Plugin

- **Key Authentication Encrypted**:
  - Fixed the `ttl` parameter. You can now set `ttl` for an encrypted key.
  - Fixed an issue where this plugin didn't accept tags.
- jwt:
  - This plugin now denies requests that have different tokens in the JWT token search locations.

    Thanks Jackson 'Che-Chun' Kuo from Latacora for reporting this issue.#9946
- mocking:
  - Fixed UUID pattern matching.
- oauth2:
  - `refresh_token_ttl` is now limited to a range between `0` and `100000000` by the schema validator. Previously, numbers that were too large caused requests to fail.#10068
- openid-connect:
  - Fixed an issue where it was not possible to specify an anonymous consumer by name.
  - Fixed an issue where the `authorization_cookie_httponly` and `session_cookie_httponly` parameters would always be set to `true`, even if they were configured as `false`.
- opentelemetry:
  - Fixed non-compliances to specification:
    - For `http.uri` in spans, the field is now the full HTTP URI. #10036
    - `http.status_code` is now present on spans for requests that have a status code. #10160
    - `http.flavor` is now a string value, not a double. #10160

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- zipkin:
  - Fixed an issue where the global plugin's sample ratio overrode the route-specific ratio.#9877
  - Fixed an issue where `trace-id` and `parent-id` strings with decimals were not processed correctly.
- datadog,**OpenTelemetry**, and **StatsD**: Fixed an issue in these plugins' batch queue processing, where metrics would be published multiple times. This caused a memory leak, where memory usage would grow without limit.#10052 #10044

Dependency

Core

- Bumped`lua-resty-openssl` from 0.8.15 to 0.8.17
- Bumped `libexpat` from 2.4.9 to 2.5.0
- Bumped `kong-openid-connect` from v2.5.0 to v2.5.2
- Bumped `openssl` from 1.1.1q to 1.1.1t
- `libyaml` is no longer built with Kong Gateway. System `libyaml` is used instead.
- Bumped `luarocks` from 3.9.1 to 3.9.2#9942
- Bumped `atc-router` from 1.0.1 to 1.0.5#9925#10143#10208
- Bumped `lua-resty-openssl` from 0.8.15 to 0.8.17#9583#10144
- Bumped `lua-kong-nginx-module` from 0.5.0 to 0.5.1#10181
- Bumped `lua-resty-session` from 3.10 to 4.0.0#10199#10230
- Bumped `libxml` from 2.10.2 to 2.10.3 to resolve CVE-2022-40303 and CVE-2022-40304

Breaking Change

Core

- The default PostgreSQL SSL version has been bumped to TLS 1.2. In `kong.conf`:
  - The default `pg_ssl_version` is now `tlsv1_2`.
  - Constrained the valid values of this configuration option to only accept the following: `tlsv1_1`, `tlsv1_2`, `tlsv1_3` or `any`.

  This mirrors the setting `ssl_min_protocol_version` in PostgreSQL 12.x and onward. See the PostgreSQL documentation for more information about that parameter.

  To use the default setting in `kong.conf`, verify that your Postgres server supports TLS 1.2 or higher versions, or set the TLS version yourself. TLS versions lower than `tlsv1_2` are already deprecated and considered insecure from PostgreSQL 12.x onward.
- Added the `allow_debug_header` configuration property to `kong.conf` to constrain the `Kong-Debug` header for debugging. This option defaults to `off`.

  If you were previously relying on the `Kong-Debug` header to provide debugging information, set `allow_debug_header: on` to continue doing so.
- **JWT plugin** (`jwt`)
  - The JWT plugin now denies any request that has different tokens in the JWT token search locations.[#9946](https: The JWT plugin now denies any request that has different tokens in the JWT token search locations.#9946
- Sessions library upgrade #10199:

l to v4.0.0. This version includes a full rewrite of the session library, and is not backwards compatible.
on, **OpenID Connect**, and **SAML**. This also affects any session configuration that uses the Session or
ng sessions for Kong Manager and Dev Portal.
g to this version. For sessions to work as expected in this version, all nodes must run Kong Gateway 3.2.x
upgrades, proxy nodes with mixed versions run for as little time as possible. During that time, the invalid

replaces `cookie_lifetime`, now has a default value of 900. Unless configured differently, sessions expire

default value of 86400. Unless configured differently, sessions expire after 86400 seconds (24 hours).
or removed. Although your configuration will continue to work as previously configured, we recommend
unexpected behavior. Refer to the upgrade guide for 3.2 for all session configuration changes and guidance
iguration.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Bugfix

Core

- Applied Nginx patch for early detection of HTTP/2 stream reset attacks. This change is in direct response to the identified vulnerability CVE-2023-44487.

Dependency

Core

- Bumped `libxml2` from 2.10.2 to 2.11.5

3.1.1.5

Release date 2023/08/25

Feature

Core

- The Redis strategy of Rate Limiting now catches connection failures.
- Added the parameter `admin_auto_create` for automatically creating a Kong admin.
- Kong Manager supports the `POST` response method for OIDC based authentication

Bugfix

Core

- Fixed an issue with the plugin iterator where sorting would become mixed up when dynamic reordering was applied. This fix ensures proper sorting behavior in all scenarios.
- Fixed an issue where `resty.dns.client` leaked UDP sockets.
- Fixed a bug where setting `anonymous_reports=false` would not silence anonymous reports.
- Fixed an issue with hybrid mode where vitals and analytics could not communicate through the cluster telemetry endpoint.
- Fixed the HTTP2 request handle in ARM artifacts.
- Fixed the OpenResty `ngx.print` chunk encoding duplicate free buffer issue that led to the corruption of chunk-encoded response data. #10816#10824
- Fixed an issue where a crashing Go plugin server process would cause subsequent requests proxied through Kong to execute Go plugins with inconsistent configurations. The issue only affects scenarios where the same Go plugin is applied to different route or service entities.
- Fixed the Dynatrace implementation.

Plugin

- oauth2:
  - Fixed a bug that `refresh_token` could be shared across instances.
- rate-limiting-advanced:

o sync the rate-limiting-advanced counters with Redis.

dcasted the wrong data in traditional cluster mode.

proxied with the wrong query parameters.

when the plugin was being used.

D

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Bumped `lua-resty-kafka` from 0.15 to 0.16

3.1.1.4

Release date 2023/05/16

Feature

Core

- Kong Manager with OIDC: Added the configuration option`admin_auto_create` to enable or disable automatic admin creation. This option is `true` by default.

Bugfix

Core

- Fixed the UDP socket leak in `resty.dns.client` caused by frequent DNS queries.#10691
- Hybrid mode: Fixed an issue where Vitals/Analytics couldn't communicate through the cluster telemetry endpoint.
- Fixed an issue where `alpine` and `ubuntu` ARM64 artifacts incorrectly handled HTTP/2 requests, causing the protocol to fail.
- Fixed the OpenResty `ngx.print` chunk encoding duplicate free buffer issue that led to the corruption of chunk-encoded response data.#10816#10824
- Fixed the Dynatrace implementation. Due to a build system issue, Kong Gateway 3.1.x packages prior to 3.1.1.4 didn't contain the debug symbols that Dynatrace requires.

Plugin

- request-transformer-advanced:
  - Fixed an issue that was causing some requests to be proxied with the wrong query parameters.

Kong Manager

- Fixed configuration fields for the StatsD plugin:
  - Added missing metric fields: `consumer_identifier`, `service_identifier`, and `workspace_identifier`.
  - Removed the non-existent `custom_identifier` field.
- Fixed an issue where the `Copy JSON` for a plugin didn't copy the full plugin configuration.
- Fixed an issue where the Zipkin plugin didn't allow the addition of `static_tags` through the Kong Manager UI.
- Added missing default values to the Vault configuration page.
- Fixed the broken Konnect link in free mode banners.
- OIDC authentication issues:
  - The `/auth` endpoint, used by Kong Manager for OIDC authentication, now correctly supports the HTTP POST method.
  - Fixed an issue with OIDC authentication in Kong Manager, where the default roles (`workspace-super-admin`, `workspace-read-only`, `workspace-portal-admin`, and `workspace-admin`) were missing from any newly created workspace.
  - Fixed an issue where users with newly registered Dev Portal accounts created through OIDC were unable to log into Dev Portal until the Kong Gateway container was restarted. This happened when `by_username_ignore_case` was set to `true`, which incorrectly caused consumers to always load from cache.

3

R

F

C

pendency from packages and images. This prevented SSL connections from using common root certificate

3

R

D

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Support for the `/vitals/reports/:entity_type` endpoint is deprecated. Use one of the following endpoints from the Vitals API instead:: :
  - For `/vitals/reports/consumer`, use `/{workspace_name}/vitals/status_codes/by_consumer` instead
  - For `/vitals/reports/service`, use `/{workspace_name}/vitals/status_codes/by_service` instead
  - For `/vitals/reports/hostname`, use `/{workspace_name}/vitals/nodes` instead

  See the Vitals documentation for more detail.

Feature

Core

- **Dev Portal** : The Dev Portal API now supports an optional `fields` query parameter on the `/files` endpoint. This parameter lets you specify which file object fields should be included in the response.
- When `router_flavor` is `traditional_compatible`, verify routes created using the Expression router instead of the traditional router to ensure created routes are actually compatible.#10088:
- `kong migrations up` now reports routes that are incompatible with the 3.0 router and stops the migration progress so that admins have a chance to adjust them.:

Bugfix

Core

- Fixed an issue where regexes generated in inso would not work in Kong Gateway.:
- Bumped `atc-router` to `1.0.2` to address the potential worker crash issue.#9927:

hybrid-mode

- Fixed an issue where Vitals data was not showing up after a license was deployed using the `/licenses` endpoint. Kong Gateway now triggers an event that allows the Vitals subsystem to be reinitialized during license preload.:
- Fixed an issue where the forward proxy between data planes and the control plane didn't support the telemetry port `8006`.:
- Reverted the removal of WebSocket protocol support for configuration sync. Backwards compatibility with 2.8.x.x data planes has been restored. #10067:

Plugin

- **mTLS Authentication**:
  - Fixed an issue where the plugin used the old route caches after routes were updated.
- oas-validation:
  - Added the OAS Validation plugin back into the bundled plugins list. The plugin is now available by default with no extra configuration necessary through `kong.conf`.
  - Fixed an issue where the plugin returned the wrong error message when failing to get the path schema spec.
  - Fixed a `500` error that occurred when the response body schema had no content field.
- rate-limiting-advanced:
  - Fixed an issue with the `local` strategy, which was not working correctly when `window_size` was set to `fixed`, and the cache would expire while the window was still valid.
- datadog **OpenTelemetry**, and **StatsD**: Fixed an issue in these plugins' batch queue processing, where metrics would be published multiple times. This caused a limit.

K

log out while redirecting to other pages or refreshing the page when OpenID Connect was used as the

d while updating the service, route, or consumer bound to a scoped plugin.:

te'] /services/*/plugins couldn't create plugins under a service.:

orkspace other than `default` would cause a `404 Not Found` error.:

ndshake Modifier and TLS Metadata Headers plugins.
ins: Kubernetes Sidecar Injector, Collector, and Upstream TLS.
lanager. This plugin must be installed manually via LuaRocks.
th plugin.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Known issues

Core

- The `ca-certificates` dependency is missing from packages and images. This prevents SSL connections from using common root certificate authorities.:

3.1.0.0

Release date 2022/12/06

Feature

Core

- You can now specify the namespaces of HashiCorp Vaults for secrets management.
- Added support for HashiCorp Vault backends to retrieve a vault token from a Kubernetes service account. See the following configuration parameters:
  - `keyring_vault_auth_method`
  - `keyring_vault_kube_role`
  - `keyring_vault_kube_api_token_file`
- FIPS 140-2 packages:
  - Kong Gateway Enterprise now provides FIPS 140-2 compliant packages for Red Hat Enterprise 8 and Ubuntu 22.04.
  - Kong Gateway FIPS distributions now support TLS connections to the PostgreSQL database.
- You can now delete consumer group configurations without deleting the group or the consumers in it.
- **Kong Manager** :
  - You can now configure the base path for Kong Manager, for example: `localhost:8445/manager`. This allows you to proxy all traffic through Kong Gateway. For example, you can proxy both API and Kong Manager traffic from one port. In addition, using the new Kong Manager base path allows you to add plugins to control access to Kong Manager. For more information, see Enable Kong Manager.
  - You can now create consumer groups in Kong Manager. This allows you to define any number of rate limiting tiers and apply them to subsets of consumers instead of managing each consumer individually. For more information, see Create Consumer Groups in Kong Manager.
  - You can now add `key-auth-enc` credentials to a consumer.
  - OpenID Connect plugin: More authorization variables have been added to the **Authorization** tab.
  - The Kong Manager overview tab has been optimized for performance.
  - You can now configure vaults for managing secrets through Kong Manager. Use the new Vaults menu to set up and manage any vaults that Kong Gateway supports. See the Vault Backends referencesfor descriptions of all the configuration options.
  - Added support for interfacing with dynamic plugin ordering.
  - Added the ability to view details about certificates.
  - Added tooltips to plugin UI with field descriptions.
  - Added support for persisting the page size of lists across pages and provided more options for page sizes.
- Allow `kong.conf` SSL properties to be stored in vaults or environment variables. Allow such properties to be configured directly as content or base64 encoded content.#9253
- Added support for full entity transformations in schemas.#9431
- The schema `map` type field can now be marked as referenceable.#9611
- Added support for dynamically changing the log level.#9744
- Added support for the `keys` and `key-sets` entities. These are used for managing asymmetric keys in various formats (JWK, PEM). For more information, see Key management.#9737

hybrid-mode

7

id mode connections. New configuration options `cluster_use_proxy`, `proxy_server` and
tion, see CP/DP Communication through a Forward Proxy.#9758#9773

P

ries. A warning will be thrown when there are too many regex routes and `router_flavor` is

educe timer usage.#9521

o

nterprise packages.

n-source and Enterprise packages.

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Extend `kong.client.tls.request_client_certificate` to support setting the Distinguished Name (DN) list hints of the accepted CA certificates.#9768

Plugin

- **Forward Proxy**:
  - `x_headers` field added. This field indicates how the plugin handles the headers`X-Real-IP`, `X-Forwarded-For`, `X-Forwarded-Proto`, `X-Forwarded-Host`, and `X-Forwarded-Port`.

    The field can take one of the following options:

    - `append`: append information from this hop in the chain to those headers. This is the default setting.
    - `transparent`: leave the headers unchanged, as if the the Kong Gateway was not a proxy.
    - `delete`: remove all the headers, as if the Kong Gateway was the originating client.

    Note that all options respect the trusted IP setting, and will ignore headers from the last hop in the chain if they are not from clients with trusted IPs.
  - `append`: append information from this hop in the chain to those headers. This is the default setting.
  - `transparent`: leave the headers unchanged, as if the the Kong Gateway was not a proxy.
  - `delete`: remove all the headers, as if the Kong Gateway was the originating client.
- **Proxy Cache Advanced**:
  - Added support for integrating with Redis clusters through the `config.redis.cluster_addresses` configuration property.
- **mTLS Authentication**:
  - Added the `config.send_ca_dn` configuration parameter to support sending CA DNs in the `CertificateRequest` message during SSL handshakes.
  - Added the `allow_partial_chain` configuration parameter to allow certificate verification with only an intermediate certificate.
- **Authentication plugins**:
  - The `anonymous` field can now be configured as the username of the consumer. This field allows you to configure a string to use as an “anonymous” consumer if authentication fails.
- acme:
  - Added support for Redis SSL, through configuration properties`config.storage_config.redis.ssl`, `config.storage_config.redis.ssl_verify`, and `config.storage_config.redis.ssl_server_name`.#9626
- app-dynamics:
  - Integrate Kong Gateway with the AppDynamics APM Platform.
- aws-lambda:
  - Added `requestContext` field into `awsgateway_compatible` input data#9380
- jwe-decrypt:
  - Allows you to decrypt an inbound token (JWE) in a request.
- mocking:
  - Added the `included_status_codes` and `random_status_code` fields. These allow you to configure the HTTP status codes for the plugin.
  - The plugin now lets you auto-generate a random response based on the schema definition without defining examples.
  - You can now control behavior or obtain a specific response by sending behavioral headers: `X-Kong-Mocking-Delay`, `X-Kong-Mocking-Example-Id`, and `X-`

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

OpenAPI 3.0 or Swagger API Specification.

field. When this field is set to true, the regex capture groups captured on the Kong Gateway route’s path nput to OPA.

- opentelemetry:
  - Added referenceable attribute to the `headers` field that could be stored in vaults.#9611
- rate-limiting-advanced:
  - Added support for deleting customer groups using the API.
  - Added `config.disable_penalty` to control whether to count `429` or not in sliding window mode.
- rate-limiting:
  - The HTTP status code and response body for rate-limited requests can now be customized. Thanks, @utix! #8930
- request-transformer-advanced:
  - Added support for navigating nested JSON objects and arrays when transforming a JSON payload.
  - The plugin now supports vault references.
- request-validator:
  - The plugin now supports the `charset` option for the`config.allowed_content_types` parameter.
- response-ratelimiting:
  - Added support for Redis SSL through configuration properties`redis_ssl` (can be set to `true` or `false`), `ssl_verify`, and `ssl_server_name`. Thanks, @dominikkukacka! #8595
- route-transformer-advanced:
  - Added the `config.escape_path` configuration parameter, which lets you escape the transformed path.
- saml:
  - Provides SAML v2.0 authentication and authorization between a service provider (Kong Gateway) and an identity provider (IdP).
- session:
  - Added new config `cookie_persistent`, which allows the browser to persist cookies even if the browser is closed. This defaults to `false` which means cookies are not persisted across browser restarts. Thanks @tschaumefor this contribution! #8187
- vault-auth:
  - Added support for KV Secrets Engine v2.
- xml-threat-protection:
  - This new plugin allows you to reduce the risk of XML attacks by checking the structure of XML payloads. This validates maximum complexity (depth of the tree), maximum size of elements and attributes.
- zipkin:
  - Added the `response_header_for_traceid` field in Zipkin plugin. The plugin sets the corresponding header in the response if the field is specified with a string value.#9173
- WebSocket service/route support was added for logging plugins: http-log
- WebSocket service/route support was added for logging plugins: file-log
- WebSocket service/route support was added for logging plugins: udp-log
- WebSocket service/route support was added for logging plugins: tcp-log
- lugins: loggly
- lugins: syslog
- lugins: kafka-log

F

C

after an update on the `user_token` field.

cept a JSON-formatted value, and was therefore unable to use vault references to secrets.

the correct log file.

n free Enterprise mode while using vaults.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Targets with a weight of `0` are no longer included in health checks, and checking their status via the `upstreams/<upstream>/health` endpoint results in the status `HEALTHCHECK_OFF`. Previously, the `upstreams/<upstream>/health` endpoint was incorrectly reporting targets with `weight=0` as `HEALTHY`, and the health check was reporting the same targets as `UNDEFINED`.
- Updated the Admin API response status code from `500` to `200` when the database is down.
- Fixed an issue when passing a license from the control plane to the data plane using the Admin API `/licenses` endpoint.
- In hybrid mode, fixed a license issue where entity validation would fail when the license entity was not processed first.
- Fixed a Websockets issue with redirects. Now, Kong Gateway redirects `ws`requests to `wss` for `wss`-only routes for parity with HTTP/HTTPS.
- **Kong Manager** :
  - Added logging for all Kong Manager access logs.
  - Fixed an issue where the **New Workspace** button was occasionally unusable.
  - Fixed the name display of plugin configurations in Kong Manager.
  - Fixed an issue where some items were missing from the suggestion list when there were many items present.
  - Removed the deprecated Vitals Reports feature from Kong Manager.
  - Fixed an issue where admins with permissions to interact with scoped entities, such as routes and services, couldn't perform operations as expected.
  - Fixed an issue where admins with the `/admins` permission were forced to log out after signing in.
  - Fixed a performance issue where admins with a large number of workspace permissions caused Kong Manager to load slowly.
- Fixed an issue where external plugins crashing with unhandled exceptions would cause high CPU utilization after the automatic restart.#9384
- Added `use_srv_name` options to upstream for balancer.#9430
- Fixed an issue in `header_filter` instrumentation where the span was not correctly created.#9434
- Fixed an issue in router building in `traditional_compatible` mode. When the field contained an empty table, the generated expression was invalid.#9451
- Fixed an issue in router rebuilding where when the `paths` field is invalid, the router's mutex is not released properly.#9480
- Fixed an issue where `kong docker-start` would fail if `KONG_PREFIX` was set to a relative path.#9337
- Fixed an issue with error-handling and process cleanup in `kong start`.#9337
- Fixed issue with prefix path normalization.#9760
- Increased the maximum request argument number of the Admin API from 100 to 1000. The Admin API now returns a `400` error if request parameters reach the limitation instead of truncating any parameters over the limit.#9510
- Paging size parameter is now propagated to next page if specified in current request.#9503

hybrid-mode

- Fixed a race condition that could cause configuration push events to be dropped when the first data plane connection was established with a control plane worker.#9616

cli

- Fixed slow CLI performance due to pending timer jobs.#9536

PDK

- Added support for `kong.request.get_uri_captures`(`kong.request.getUriCaptures`)#9512
- Fixed parameter type of `kong.service.request.set_raw_body`(`kong.service.request.setRawBody`), return type of`kong.service.response.get_raw_body`(`kong.service.request.getRawBody`), and body parameter type of `kong.response.exit` to bytes. Note that the old version of the go PDK is incompatible after this change.#9526

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

way connects to Redis SSL port `6379` with `config.ssl=false`.

vironment variables in ECS environment.#9460

eld in lambda output now results in a more obvious error log entry with a `502` code.#9598

ld fail in the following situations:

ad no service.
ue.#9177

service.

  - The route's associated service had a `path` value.#9177

- http-log:
  - Fixed an issue where queue ID serialization did not include `queue_size` and `flush_timeout`.#9789
- mocking:
  - Fixed an issue with `accept` headers not being split and not working with wildcards. The `;q=` (q-factor weighting) of `accept` headers is now supported.
- opa:
  - Removed redundant deprecated code from the plugin.
- opentelemetry:
  - Fixed an issue that the default propagation header was not configured to `w3c` correctly.#9457
  - Replaced the worker-level table cache with`BatchQueue` to avoid data race.#9504
  - Fixed an issue that the `parent_id` was not set on the span when propagating w3c traceparent.#9628
- rate-limiting-advanced:
  - The plugin now ensures that shared dict TTL is higher than `config.sync_rate`, otherwise Kong Gateway would lose all request counters in shared dict.
- request-termination:
  - The plugin no longer allows setting `status_code` to `null`.#9400
- request-transformer:
  - Fixed a bug when header renaming would override the existing header and cause unpredictable results.#9442
- response-transformer:
  - Fixed the bug that the plugin would break when receiving an unexpected body.#9463
- zipkin:
  - Fixed an issue where Zipkin plugin couldn't parse OT baggage headers due to an invalid OT baggage pattern.#9280
- Added the missing `protocols` field to the following plugin schemas: Azure Functions (`azure-functions`)
- Added the missing `protocols` field to the following plugin schemas: gRPC Gateway (`grpc-gateway`)
- Added the missing `protocols` field to the following plugin schemas: gRPC Web (`grpc-web`)
- Added the missing `protocols` field to the following plugin schemas: Serverless pre-function (`pre-function`)
- Added the missing `protocols` field to the following plugin schemas: Prometheus (`prometheus`)
- Added the missing `protocols` field to the following plugin schemas: Proxy Caching (`proxy-cache`)
- Added the missing `protocols` field to the following plugin schemas: Request Transformer (`request-transformer`)
- Added the missing `protocols` field to the following plugin schemas: Session (`session`)
- Added the missing `protocols` field to the following plugin schemas: Zipkin (`zipkin`)

Known Issues

C

u will still see `info` and `error` entries in the logs.

E

h

ed in favor of the wRPC protocol introduced in 3.0.0.0. Rolling upgrades from 2.8.x.y to 3.1.0.0 are not
can perform a rolling upgrade to 3.1.0.0. For more information, see Upgrade Kong Gateway 3.1.x.#9740

3

R

F

P

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- request-transformer-advanced:
  - Values stored in `key:value` pairs in this plugin's configuration are now referenceable, which means they can be stored as secrets in a vault.

Bugfix

Core

- Unpinned the version of `alpine` used in the `kong/kong-gateway` Docker image. Previously, the version was pinned to 3.10, which was creating outdated `alpine` builds.:
- Fixed an issue with how Kong initializes `resty.events`. The code was previously using `ngx.config.prefix()` to determine the listening socket path to provide to the resty.events module. This caused breakage when Nginx was started with a relative path prefix. This meant that you couldn't start 3.0.x with the same default configuration as 2.8.x.:
- Fixed an issue with secret management references for HashiCorp Vault. By default, Kong passes secrets to the Nginx using environment variables when using `kong start`. Nginx was being started directly without calling `kong start`, so the secrets were not available at initialization. #9478:
- Fixed the Amazon Linux RPM installation instructions.:

Kong Manager

- Removed the endpoint `all_routes` from configurable RBAC endpoint permissions. This endpoint was erroneously appearing in the endpoints list, and didn't configure anything.:
- Fixed an issue that allowed unauthorized IDP users to log in to Kong Manager. These users had no access to any resources in Kong Manager, but were able to go beyond the login screen.:
- Fixed an issue where, in an environment with a valid Enterprise license, admins with no access to the `default` workspace would see a message prompting them to upgrade to Kong Enterprise.:
- Fixed pagination issues with Kong Manager tables.:
- Fixed broken `Learn more` links.:
- Fixed an issue with group to role mapping, where it didn't support group names with spaces.:
- Fixed the Cross Site Scripting (XSS) security vulnerability in the Kong Manager UI.:
- Fixed an RBAC issue where permissions applied to specific endpoints (for example, an individual service or route) were not reflected in the Kong Manager UI.:
- Removed New Relic from Kong Manager. Previously, `VUE_APP_NEW_RELIC_LICENSE_KEY` and`VUE_APP_SEGMENT_WRITE_KEY` were being exposed in Kong Manager with invalid values.:
- Removed the action dropdown menu on service and route pages for read-only users.:
- Fixed the **Edit Configuration** button for Dev Portal applications.:
- Fixed an RBAC issue where the roles page listed deleted roles.:
- Fixed an issue where the orphaned roles would remain after deleting a workspace and cause the **Teams** > **Admins** page to break.:
- Added the missing **Copy JSON** button for plugin configuration.:
- Fixed an issue where the **New Workspace** button on the global workspace dashboard wasn't clickable on the first page load.:
- Removed the ability to add multiple documents per service from the UI. Each service only supports one document, so the UI now reflects that.:
- The Upstream Timeout plugin now has an icon and is part of the Traffic Control category.:

e ACL credentials from the consumer credentials list. This happened because the name of the plugin, `acl`,

disabling the Dev Portal for a workspace wouldn't change the Kong Manager menu.:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ou can change a plugin's static priority by specifying the order in which plugins run. This lets you run gins.

replaces the primary library, OpenSSL, with BoringSSL, which at its core uses the FIPS 140-2 compliant

supported in Ubuntu 20.04.

Note : The Kong Gateway FIPS package is not currently compatible with SSL connections to PostgreSQL.

- Kong Gateway now includes WebSocket validation functionality. Websockets are a type of persistent connection that works on top of HTTP.

  Previously, Kong Gateway 2.x supported limited WebSocket connections, where plugins only ran during the initial connection phase instead of for each frame. Now, Kong Gateway provides more control over WebSocket traffic by implementing plugins that target WebSocket frames.

  This release includes:

  - Service and route support for `ws` and `wss` protocols
  - Two new plugins: WebSocket Size Limit and WebSocket Validator
  - WebSocket plugin development capabilities (**Beta feature**)
    - PDK modules: kong.websocket.client and kong.websocket.upstream
    - New plugin handlers

  Learn how to develop WebSocket plugins with our plugin development guide.

- In this release, Kong Manager ships a with a refactored design and improved user experience.

  Notable changes:

  - Reworked workspace dashboards, both for specific workspaces and at the multi-workspace level.
  - License metrics now appear at the top of overview pages.
  - Restructured the layout and navigation to make workspace selection a secondary concern.
  - Grayed out portal buttons when you don't have permissions.
  - Added license level to phone home metrics.
  - Added more tooltips.

- Secrets management is now generally available.

  - Added GCP integration support for the secrets manager. GCP is now available as a vault backend.
  - The `/vaults-beta` entity has been deprecated and replaced with the `/vaults` entity.#8871#9217

- Kong Gateway now provides slim and UBI images. Slim images are docker containers built with a minimal set of installed packages to run Kong Gateway. From 3.0 onward, Kong Docker images will only contain software required to run the Gateway. This ensures that false positive vulnerabilities don't get flagged during security scanning.

  If you want to retain or add other dependencies, you can build custom Kong Docker images.

- The base OS for our convenience docker tags (for example, `latest`, `3.0.0.0`, `3.0`) has switched from Alpine to Debian.

- Added key recovery for keyring encryption. This exposes a new endpoint for the Admin API, `/keyring/recover`, and requires `keyring_recovery_public_key` to be set in `kong.conf`.

- You can now encrypt declarative configuration files on data planes in DB-less and hybrid modes using AES-256-GCM or chacha20-poly1305 encryption algorithms.

  Set your desired encryption mode with the `declarative_config_encryption_mode` configuration parameter.

- This release introduces a new router implementation: `atc-router`. This router is written in Rust, a powerful routing language that can handle complex routing requirements. The new router can be used in traditional-compatible mode, or use the new expression-based language.

  With the new router, we have:

  - Reduced router rebuild time when changing Kong's configuration
  - Increased runtime performance when routing requests
  - Reduced P99 latency from 1.5s to 0.1s with 10,000 routes

  Learn more about the router:

  - Configure routes using expressions
  - Router Expressions Language reference

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

n.#8179

nTelemetry API specs, and adds built-in instrumentations.

porter plugin. Built-in instrumentation types and sampling rate are configurable through
g_sampling_rate options.#8724

stream `hash_on`for load balancing.#8701

s to replace shared memory-based `lua-resty-worker-events`.#8890

ts you specify a table name. Previously, the name was deduced by the entity `name` attribute.#9182

255

were not needed. Now when a target is removed or updated, the DNS record associated with it is removed

- Improved error handling and debugging info in the DNS code.#8902

- Kong Gateway will now attempt to recover from an unclean shutdown by detecting and removing dangling Unix sockets in the prefix directory.#9254
- A new CLI command, `kong migrations status`, generates the migration status in a JSON file.
- Removed the warning for `AAAA` being experimental with `dns_order`.

Performance

- Kong Gateway does not register unnecessary event handlers on hybrid mode control plane nodes anymore. #8452.
- Use the new timer library to improve performance, except for the plugin server.#8912
- Increased the use of caching for DNS queries by activating `additional_section` by default.#8895
- `pdk.request.get_header` has been changed to a faster implementation. It doesn't fetch all headers every time it's called.#8716
- Conditional rebuilding of the router, plugins iterator, and balancer on data planes.#8519,#8671
- Made configuration loading code more cooperative by yielding.#8888
- Use the LuaJIT encoder instead of JSON to serialize values faster in LMDB.#8942
- Made inflating and JSON decoding non-concurrent, which avoids blocking and makes data plane reloads faster.#8959
- Stopped duplication of some events.#9082
- Improved performance of configuration hash calculation by using `string.buffer` and `tablepool`.#9073
- Reduced cache usage in DB-less mode by not using the Kong cache for routes and services in LMDB.#8972

Admin API

- Added a new `/timers` Admin API endpoint to get timer statistics and worker info.#8912#8999
- The `/` endpoint now includes plugin priority.#8821

hybrid-mode

- Added wRPC protocol support. Configuration synchronization now happens over wRPC. wRPC is an RPC protocol that encodes with ProtoBuf and transports with WebSocket.#8357
  - To keep compatibility with earlier versions, added support for the control plane to fall back to the previous protocol to support older data planes.#8834
  - Added support to negotiate services supported with wRPC protocol. We will support more services in the future.#8926
- Declarative configuration exports now happen inside a transaction in PostgreSQL.#8586

Plugin

- **mTLS Authentication**:
  - Introduced certificate revocation list (CRL) and OCSP server support with the following parameters: `http_proxy_host`, `http_proxy_port`, `https_proxy_host`, and `https_proxy_port`.
- acme:
  - Added the `allow_any_domain` field. It defaults to false and if set to true, the gateway will ignore the `domains` field.#9047
- aws-lambda:

h the `aws_assume_role_arn` and`aws_role_session_name` configuration parameters.#8900

a valid return when working in proxy integration mode.#8765

the IAM role ARN.#8907

the `graphql_server_path` configuration parameter.

on mechanism.

up membership. The new configuration parameter, `groups_required`, is an array of string elements that
e request to be authorized.

d.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- opa:
  - New configuration parameter `include_body_in_opa_input`: When enabled, include the raw body as a string in the OPA input at `input.request.http.body` and the body size at `input.request.http.body_size`.
  - New configuration parameter `include_parsed_json_body_in_opa_input`: When enabled and content-type is `application/json`, the parsed JSON will be added to the OPA input at `input.request.http.parsed_body`.
- prometheus:
  - `nginx_http_current_connections` and `nginx_stream_current_connections` were merged into to `nginx_hconnections_total` (or `nginx_current_connections`?)
  - `request_count` and `consumer_status` were merged into http_requests_total.

    If the `per_consumer` config is set to `false`, the `consumer` label will be empty. If the `per_consumer` config is `true`, the `consumer` label will be filled.
  - Updated the Grafana dashboard that comes packaged with Kong
- statsd:
  - **Newly open-sourced plugin capabilities** : All capabilities of the StatsD Advanced plugin are now bundled in the StatsD plugin.#9046
- zipkin:
  - Added support for including the HTTP path in the span name with the`http_span_name` configuration parameter.#8150
  - Added support for socket connect and send/read timeouts through the `connect_timeout`, `send_timeout`, and `read_timeout` configuration parameters. This can help mitigate`ngx.timer` saturation when upstream collectors are unavailable or slow.#8735
- Starting with version 3.0, all bundled plugin versions are the same as the Kong Gateway version.#8772:

  Plugin documentation now refers to the Kong Gateway version instead of the individual plugin version.
- New plugins:
  - opentelemetry: Export tracing instrumentations to any OTLP/HTTP compatible backend.`opentelemetry_tracing` configuration must be enabled to collect the core tracing spans of Kong Gateway.#8826
  - **TLS Handshake Modifier**: Make certificates available to other plugins acting on the same request.
  - **TLS Metadata Headers**: Proxy TLS client certificate metadata to upstream services via an HTTP headers.
  - **WebSocket Size Limit**: Allows operators to specify a maximum size for incoming WebSocket messages.
  - **WebSocket Validator**: Validate individual WebSocket messages against a user-specified schema before proxying them.
- Collector (`collector`): The deprecated Collector plugin has been removed.

Configuration

- You can now configure `openresty_path` to allow developers and operators to specify the OpenResty installation to use when running Kong Gateway, instead of using the system-installed OpenResty.#8412
- Added `ipv6only` to listen options `admin_listen`, `proxy_listen`, and `stream_listen`.#9225
- Added `so_keepalive` to listen options `admin_listen`, `proxy_listen`, and `stream_listen`.#9225
- Add LMDB DB-less configuration persistence and removed the JSON-based configuration cache for faster startup time.#8670
- `nginx_events_worker_connections=auto` now has a lower bound of 1024.#9276
- `nginx_main_worker_rlimit_nofile=auto` now has a lower bound of 1024.#9276

PDK

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`me()`. This function returns the request start time, in Unix epoch milliseconds.#8688

`.id` if nothing from `cache_key` is found.#8553

plane would crash if any errors occurred during the initialization of the keyring module.

ing keys after a soft reload.

nd page while iterating over a foreign key field using the DAO.#9255

arting a control plane.

- **Vitals** :

  - Disabled `phone_home` for `anonymous_reports` on the data plane.
  - The Kong Gateway version information is now sent in the telemetry request query parameter.

- **Kong Manager** :

  - Fixed the workspace dashboard's loading state. Previously, a dashboard with no request data and an existing service would still prompt users to add a service.
  - Fixed an issue where Kong Manager allowed selection of metrics not supported by the Datadog plugin.
  - Fixed the values accepted for upstream configuration in Kong Manager. Previously, fields that were supposed to accept decimals would only accept whole numbers.
  - Fixed an issue where you couldn't save or update `pre-function` plugin configuration when the updated value contained a comma (`,`).
  - The service name field on the Service Contracts page now correctly shows the service display name. Previously, it showed the service ID.
  - Fixed an issue where, after updating the CA certificate, the page wouldn't return to the certificate view.
  - Fixed an issue where the port was missing from the service URL on the service overview page.
  - Fixed an issue where switching between workspace dashboard pages would not update the Dev Portal URL.
  - Fixed issues with plugins:
    - The Exit Transformer plugin can now load Lua functions added through Kong Manager.
    - The CORS plugin now treats regexes properly for the `config.origins` field.
    - The Datadog plugin now accepts an array for the `tags` field. Previously, it was incorrectly expecting a string.
  - Fixed an `HTTP 500` error that occurred when sorting routes by the **Hosts** column, then clicking **Next** on a paginated listing.
  - Fixed an issue that prevented developer role assignments from displaying in Kong Manager. When viewing a role under the Permissions tab in the Dev Portal section, the list of developers wouldn't update when a new developer was added. Kong Manager was constructing the wrong URL when retrieving Dev Portal assignees.
  - Fixed an issue where admins couldn't switch workspaces if they didn't have an roles in the default workspace.
  - Fixed a display issue with Dev Portal settings in Kong Manager.
  - Improved the error that appeared when trying to view admin roles without permissions for the resource. Instead of displaying `404 workspace not found`, the error now informs the user that they don't have access to view roles.

- Fixed an issue where the data plane would reload and lose its license after an Nginx reload.

- Fixed issues in dependencies:

  - `kong-gql`: Fixed variable definitions to handle non-nullable/list-type variables correctly.
  - `lua-resty-openssl-aux-module`: Fixed an issue with getting `SSL_CTX` from a request.

- The schema validator now correctly converts `null` from declarative configurations to `nil`.#8483

- Kong now reschedules router and plugin iterator timers only after finishing the previous execution, avoiding unnecessary concurrent executions.#8567

- External plugins now handle returned JSON with null member correctly.#8611

- Fixed an issue where the address of an environment variable could change but the code didn't check that it was fixed after init.#8581

- Fixed an issue where the Go plugin server instance would not be updated after a restart.#8547

- Fixed an issue on trying to reschedule the DNS resolving timer when Kong was being reloaded.#8702

- The private stream API has been rewritten to allow for larger message payloads.#8641

- Fixed an issue that the client certificate sent to the upstream was not updated when using the `PATCH` method.#8934

- Fixed an issue where the control plane and wRPC module interaction would cause Kong to crash when calling `export_deflated_reconfigure_payload` without a `pcall`.#8668

- Moved all `.proto` files to `/usr/local/kong/include` and ordered by priority.#8914

- Fixed an issue that caused unexpected 404 errors when creating or updating configs with invalid options.#8831

- Fixed an issue that caused crashes when calling some PDK APIs.#8604

- Fixed an issue that caused crashes when go PDK calls returned arrays.#8891

- Plugin servers now shutdown gracefully when Kong exits.#8923

not take `y` as default.#9114

t connect to Cassandra on init.#8847

in the `off` strategy.#9174

ess_auto_fields.#9049

timers to keep track of upstreams when `worker_consistency = eventual`.#8694,#8858

using only a hostname for targets set only by their hostname.#8797

not being properly invalidated after a cascade delete.#9261

ning no longer overwrites the existing `.kong_env` file #9254

tatus.#8690

- Fixed an issue where the Admin API didn't display `Allow` and `Access-Control-Allow-Methods` headers with `OPTIONS` requests.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Plugin

- **Forward Proxy**:
  - Fixed a proxy authentication error caused by incorrect base64 encoding.
  - Use lowercase when overwriting the Nginx request host header.
  - The plugin now allows multi-value response headers.
- **HMAC Authentication**:
  - Removed deprecated signature format using `ngx.var.uri`.#8558
- **gRPC Gateway**:
  - Fixed the handling of boolean fields from URI arguments.#9180
- acme:
  - The default value of the `auth_method` configuration parameter is now set to `token`.#8565
  - Added a cache for `domains_matcher`.#9048
- aws-lambda:
  - Removed the deprecated `proxy_scheme` field from the plugin's schema.#8566
  - Changed the path from `request_uri` to `upstream_uri` to fix an issue where the URI could not follow a rule defined by the Request Transformer plugin.#9058 #9129
- http-log:
  - Log output is now restricted to the workspace the plugin is running in. Previously, the plugin could log requests from outside of its workspace.
- ldap-auth-advanced:
  - Fixed an issue where Kong Manager LDAP authentication failed when `base_dn` was the domain root.
- ldap-auth:
  - Refactored ASN.1 parser using OpenSSL API through FFI.#8663
- mocking:
  - Fixed an issue where `204` responses were not handled correctly and you would see the following error:`"No examples exist in API specification for this resource"`.
  - `204` response specs now support empty content elements.
- openid-connect:
  - Fixed an issue with `kong_oauth2` consumer mapping.
- pre-function:
  - Fixed a problem that could cause a crash.#9269
- rate-limiting-advanced:
  - Fixed error handling when calling `get_window` and added more buffer on the window reserve.
  - ion when in hybrid or DB-less mode and strategy is set to `cluster`.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

hen the `cluster` policy was used with two or more metrics (for example, `second` and `day`.)#8968

r.#8564

nnection time from Nginx to the upstream.#8848

me.#9230

eader spec, which makes `parent_id` optional.#8352

orting based on their name.#8957

tter when a plugin instance loses the `instances_id` in an event handler.#8652

Clustering

- The cluster listener now uses the value of `admin_error_log` for its log file instead of `proxy_error_log`.#8583
- Fixed a typo in some business logic that checks the Kong role before setting a value in cache at startup. #9060
- Fixed an issue in hybrid mode where, if a service was set to `enabled: false` and that service had a route with an enabled plugin, any new data planes would receive empty configuration.#8816
- Localized `config_version` to avoid a race condition from the new yielding config loading code.#8188

PDK

- `kong.response.get_source()` now returns an error instead of an exit when plugin throws a runtime exception in the access phase.#8599
- `kong.tools.uri.normalize()` now escapes reserved and unreserved characters more accurately.#8140
- RFC3987 validation on route paths was removed, allowing operators to create a route with an invalid path URI like `/something|` which can not match any incoming request. This validation will be added back in a future release.

Dependency

Core

- Bumped `openresty` from 1.19.9.1 to 1.21.4.1#8850
- Bumped `pgmoon` from 1.13.0 to 1.15.0#8908#8429
- Bumped `openssl` from 1.1.1n to 1.1.1q#9074#8544#8752#8994
- Bumped `resty.openssl` from 0.8.8 to 0.8.10#8592#8753#9023
- Bumped `inspect` from 3.1.2 to 3.1.3#8589
- Bumped `resty.acme` from 0.7.2 to 0.8.1#8680#9165
- Bumped `luarocks` from 3.8.0 to 3.9.1#8700#9204
- Bumped `luasec` from 1.0.2 to 1.2.0#8754#8754
- Bumped `resty.healthcheck` from 1.5.0 to 1.6.1#8755#9018#9150
- Bumped `resty.cassandra` from 1.5.1 to 1.5.2#8845
- Bumped `penlight` from 1.12.0 to 1.13.1#9206
- Bumped `lua-resty-mlcache` from 2.5.0 to 2.6.0#9287
- Bumped `lodash` for Dev Portal from 4.17.11 to 4.17.21
- Bumped `lodash` for Kong Manager from 4.17.15 to 4.17.21

Breaking Change

Plugin

- **Deprecated** :
  - The StatsD Advanced plugin has been deprecated and will be removed in 4.0. All capabilities are now available in the StatsD plugin.
- acl:

t configuration parameters. #8560

n parameter is now `token`.

ough the plugin configuration with the `aws_region` field parameter, or with environment variables.

s to be set at the same time, and always applies the SigV4 signature.#8082

r header name, where it previously took an array of values.#6992

nx context (`ngx.ctx.authenticated_jwt_token`). Custom plugins which depend on that value being set
red context instead (`kong.ctx.shared.authenticated_jwt_token`) before upgrading to 3.0.

- pre-function:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Removed the deprecated `config.functions` configuration parameter from the Serverless Functions plugins' schemas. Use the `config.access` phase instead.#8559

- prometheus:
  - High cardinality metrics are now disabled by default.
  - Decreased performance penalty to proxy traffic when collecting metrics.
  - `http_status` to `http_requests_total`.
  - `latency` to `kong_request_latency_ms` (HTTP), `kong_upstream_latency_ms`, `kong_kong_latency_ms`, and `session_duration_ms` (stream).

    Kong latency and upstream latency can operate at orders of different magnitudes. Separate these buckets to reduce memory overhead.
  - `kong_bandwidth` to `kong_bandwidth_bytes`.
  - Removed the following metric: `http_consumer_status`
  - New metrics:
    - `session_duration_ms`: monitoring stream connections.
    - `node_info`: Single gauge set to 1 that outputs the node's ID and Kong Gateway version.
  - `session_duration_ms`: monitoring stream connections.
  - `node_info`: Single gauge set to 1 that outputs the node's ID and Kong Gateway version.
  - `http_requests_total` has a new label, `source`. It can be set to `exit`, `error`, or `service`.
  - All memory metrics have a new label: `node_id`.
  - The following metric names were adjusted to add units to standardize where possible:
    - `http_status` to `http_requests_total`.
    - `latency` to `kong_request_latency_ms` (HTTP), `kong_upstream_latency_ms`, `kong_kong_latency_ms`, and `session_duration_ms` (stream).

    Kong latency and upstream latency can operate at orders of different magnitudes. Separate these buckets to reduce memory overhead.
    - `kong_bandwidth` to `kong_bandwidth_bytes`.
    - `nginx_http_current_connections` and `nginx_stream_current_connections` were merged into to `nginx_connections_total`.
    - `request_count` and `consumer_status` were merged into `http_requests_total`.

    If the `per_consumer` config is set to `false`, the `consumer` label will be empty. If the `per_consumer` config is `true`, the `consumer` label will be filled.
  - `nginx_http_current_connections` and `nginx_stream_current_connections` were merged into to `nginx_connections_total`.
  - `request_count` and `consumer_status` were merged into `http_requests_total`.

    If the `per_consumer` config is set to `false`, the `consumer` label will be empty. If the `per_consumer` config is `true`, the `consumer` label will be filled.
  - Updated the Grafana dashboard that comes packaged with Kong#8712
  - The plugin doesn't export status codes, latencies, bandwidth and upstream health check metrics by default. They can still be turned on manually by setting `status_code_metrics`,`lantency_metrics`, `bandwidth_metrics` and `upstream_health_metrics` respectively.#9028

- proxy-cache:
  - These plugins don't store response data in `ngx.ctx.proxy_cache_hit` anymore. Logging plugins that need the response data must now read it from `kong.ctx.shared.proxy_cache_hit`.#8607

- rate-limiting:

modes.#9344

s a `service.` prefix: `kong.service.<service_identifier>.request.count`.

equest.status.<status>` has been renamed to `kong.service.<service_identifier>.status.`

ser.<consumer_identifier>.request.status.<status>` has been renamed to `kong.service.
dentifier>.status.<status>`.

t.status.<status>` has been renamed to `kong.service.<service_identifier>.status.<status>`.

consumer_identifier>.request.status.<status>` has been renamed to `kong.service.
fier>.status.<status>`.

cs `status_count` and `status_count_per_user` has been removed.

oading order is explicit. Loading them in a hash-like table is no longer supported.#8988

VERSION` ("x.y.z" format) field in their `handler.lua` file, otherwise the plugin will fail to load.#8836

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- The old `kong.plugins.log-serializers.basic` library was removed in favor of the PDK function `kong.log.serialize`. Upgrade your plugins to use the PDK.#8815
- The support for deprecated legacy plugin schemas was removed. If your custom plugins still use the old (`0.x era`) schemas, you are now forced to upgrade them.#8815
- Updated the priority for some plugins.: `acme` changed from `1007` to `1705`
- Updated the priority for some plugins.: `basic-auth` changed from `1001` to `1100`
- Updated the priority for some plugins.: `canary` changed from `13` to `20`
- Updated the priority for some plugins.: `degraphql` changed from `1005` to `1500`
- Updated the priority for some plugins.: `graphql-proxy-cache-advanced` changed from `100` to `99`
- Updated the priority for some plugins.: `hmac-auth` changed from `1000` to `1030`
- Updated the priority for some plugins.: `jwt` changed from `1005` to `1450`
- Updated the priority for some plugins.: `jwt-signer` changed from `999` to `1020`.
- Updated the priority for some plugins.: `key-auth` changed from `1003` to `1250`
- Updated the priority for some plugins.: `key-auth-advanced` changed from `1003` to `1250`
- Updated the priority for some plugins.: `ldap-auth` changed from `1002` to `1200`
- Updated the priority for some plugins.: `ldap-auth-advanced` changed from `1002` to `1200`
- Updated the priority for some plugins.: `mtls-auth` changed from `1006` to `1600`
- Updated the priority for some plugins.: `oauth2` changed from `1004` to `1400`
- Updated the priority for some plugins.: `openid-connect` changed from `1000` to `1050`
- Updated the priority for some plugins.: `rate-limiting` changed from `901` to `910`
- Updated the priority for some plugins.: `rate-limiting-advanced` changed from `902` to `910`
- Updated the priority for some plugins.: `route-by-header` changed from `2000` to `850`
- Updated the priority for some plugins.: `route-transformer-advanced` changed from `800` to `780`
- Updated the priority for some plugins.: `pre-function` changed from `+inf` to `1000000`
- Updated the priority for some plugins.: `vault-auth` changed from `1003` to `1350`
- Kong plugins no longer support `CREDENTIAL_USERNAME` (`X-Credential-Username`). Use the constant `CREDENTIAL_IDENTIFIER` (`X-Credential-Identifier`) when setting the upstream headers for a credential.#8815

deployment

- Deprecated and stopped producing Amazon Linux 1 containers and packages. Amazon Linux 1 reached end-of-life on December 31, 2020.Kong/docs.konghq.com #3966
- Deprecated and stopped producing Debian 8 (Jessie) containers and packages. Debian 8 reached end-of-life in June 30, 2020.Kong/kong-build-tools #448

Core

- As of 3.0, Kong Gateway's schema library's `process_auto_fields` function will not make deep copies of data that is passed to it when the given context is `select`. where we believe the data most of the time comes from a driver like `pgmoon` or `lmdb`.

verriding the given table, it must make its own copy before passing it to the function now.#8796

) schemas was removed in favor of the typed `shorthand_fields`. If your custom schemas still use fields.#8815

oved from Kong schemas and Kong field schemas.#8958

oved. If your custom Nginx templates still use it, change it to `Kong.admin_content`.#8815

oved in favor of the PDK `kong.*`.#8874

uration persistence is now done automatically with LMDB.#8704

x.ctx.balancer_data.#9043

ong Gateway now stores the unnormalized path, but the regex path always pattern-matches with the cent-encoding in the regex path pattern to ensure different forms of URI matches. That is no longer c3986, write all other characters without percent-encoding.#9024

r a `route.path` is a regex pattern. From 3.0 onward, all regex paths must start with the `"~"` prefix, and all paths that don't start with `"~"` will be considered plain text. The migration process should automatically convert the regex paths when upgrading from 2.x to

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

3.0.#9027

- Bumped the version number (`_format_version`) of declarative configuration to `3.0` for changes on `route.path`. Declarative configurations using older versions are upgraded to `3.0` during migrations.

  > **Do not sync (`deck sync`) declarative configuration files from 2.8 or earlier to 3.0.** Old configuration files will overwrite the configuration and create compatibility issues. To grab the updated configuration, `deck dump` the 3.0 file after migrations are completed.

  #9078

- Tags may now contain space characters.#9143
- Support for the `nginx-opentracing` module is deprecated as of `3.0` and will be removed from Kong in `4.0` (see the Known Limitations section for additional information).
- We removed regex look-around and backreferences support in the the atc-router. These are rarely used features and removing support for them improves the speed of our regex matching. If your current regexes use look-around or backreferences you will receive an error when attempting to start Kong, showing exactly what regex is incompatible. Users can either switch to the `traditional` router flavor or change the regex to remove look-around / backreferences.

Admin API

- The Admin API endpoint `/vitals/reports` has been removed.
- `POST` requests on `/targets` endpoints are no longer able to update existing entities. They are only able to create new ones.#8596,#8798. If you have scripts that use`POST` requests to modify `/targets`, change them to `PUT`requests to the appropriate endpoints before updating to Kong Gateway 3.0.
- Insert and update operations on duplicated targets return a `409` error.#8179,#8768
- The list of reported plugins available on the server now returns a table of metadata per plugin instead of a boolean `true`.#8810

PDK

- The `kong.request.get_path()` PDK function now performs path normalization on the string that is returned to the caller. The raw, non-normalized version of the request path can be fetched via `kong.request.get_raw_path()`.#8823
- `pdk.response.set_header()`, `pdk.response.set_headers()`, `pdk.response.exit()` now ignore and emit warnings for manually set `Transfer-Encoding` headers.#8698
- The PDK is no longer versioned.#8585
- The JavaScript PDK now returns `Uint8Array` for `kong.request.getRawBody`,`kong.response.getRawBody`, and `kong.service.response.getRawBody`. The Python PDK returns `bytes` for `kong.request.get_raw_body`,`kong.response.get_raw_body`, and `kong.service.response.get_raw_body`. Previously, these functions returned strings.#8623
- The `go_pluginserver_exe` and `go_plugins_dir` directives are no longer supported. #8552. If you are using Go plugin server, migrate your plugins to use the Go PDK before upgrading.

Configuration

- The Kong constant `CREDENTIAL_USERNAME` with the value of `X-Credential-Username` has been removed.#8815
- The default value of `lua_ssl_trusted_certificate` has changed to `system`#8602 to automatically load the trusted CA list from the system CA store.
- It is no longer possible to use a `.lua` format to import a declarative configuration file from the `kong`CLI tool. Only JSON and YAML formats are supported. If your update procedure with Kong Gateway involves executing `kong config db_import config.lua`, convert the `config.lua` file into a `config.json` or `config.yml` file before upgrading.#8898
- The data plane config cache mechanism and its related configuration options (`data_plane_config_cache_mode` and `data_plane_config_cache_path`) have been removed in favor of LMDB.

n

migrations) is no longer supplied with Kong Gateway.#8781

assandra migrations, designating a function to call. The `up_f` part is invoked after the `up` part has been assandra.

K

C

eatures:

ot supported.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - This is a known issue planned to be fixed in the next 2.8 release. If this is a requirement for upgrading, Kong operators should upgrade to that version before beginning a upgrade to 3.0.0.0.
  - See Upgrade Kong Gateway for more details.

- OpenTracing: There is an issue with `nginx-opentracing` in this release, so it is not recommended to upgrade yet if you are an OpenTracing user. This will be rectified in an upcoming patch/minor release.
- The Kong Gateway FIPS package is not currently compatible with SSL connections to PostgreSQL.

2.8.4.14

Release date 2025/04/14

Feature

Core

- Added an option for GitHub Actions to build nginx/OpenResty with debug symbols.

CLI Command

- Added a `--lts_34_compatibility` option to the `check` command to perform configuration compatibility check for upgrading to version 3.4.x.x.

Bugfix

Core

- Disabled requests with both Content-Length and Transfer-Encoding by backporting the Nginx patch.

Dependency

Core

- Pinned `LPEG` to `1.1.0` to keep the version consistent across all active branches. This is not a version bump, it is just pinning the version already in use.

2.8.4.13

Release date 2024/09/20

Breaking Change

Core

- Fixed RPM relocation by setting the default prefix to `/`, and added a symbolic link for `resty` to handle missing `/usr/local/bin` in `PATH`.

Bugfix

Core

- Fixed an issue where luarocks-admin was not available in /usr/local/bin.

Plugin

ing due to race condition.

2

R

D

C

ife (EOL) dates on June 30, 2024. As of this patch, Kong is not building Kong Gateway 2.8.x installation
ong is no longer providing official support for any Kong version running on these systems.

D

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Made the RPM package relocatable with the default prefix set to `/`.

2.8.4.11

Release date 2024/06/22

Bugfix

Core

- Reverted DNS client to original behaviour of ignoring ADDITIONAL SECTION in DNS responses.

2.8.4.10

Release date 2024/06/18

Feature

Core

- add a Docker image for RHEL8.

Bugfix

Core

- **Vitals**: Fixed a bug that each data plane connecting to the control plane would trigger the control plane to create a redundant table rotater timer.

Plugin

- rate-limiting-advanced Refactored `kong/tools/public/rate-limiting` to keep the original interfaces unchanged (backward compatibility) and extend a new interface `new_instance` to provide isolation between different plugins. If you are using custom Rate Limiting plugins based on this library, please update the initialization code to the new format like 'local ratelimiting = require("kong.tools.public.rate-limiting").new_instance("custom-plugin-name")'. The old interface will be removed in the upcoming major release.

Dependency

Core

- Improve the robustness of lua-cjson when handling unexpected input.

2.8.4.9

Release date 2024/04/19

Bugfix

PDK

urn a number which was caused by an incorrectly stored string value in ngx.ctx.host_port.

C

d cached even when configurations were changed.

P

being correctly parsed and coerced into their defined types.

2

R

F

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Configuration

- now TLSv1.1 and lower is by default disabled in OpenSSL 3.x

Bugfix

Configuration

- Fix an issue where an external plugin (Go, Javascript, or Python) would fail to apply a change to the plugin config via the Admin API.
- Set security level of gRPC's TLS to 0 when ssl_cipher_suite is set to old

Core

- update file permission of kong.logrotate to 644
- Fix the missing router section for the output of the request-debugging
- Fix a bug that `/metrics` endpoint throws an error when database is down.
- Fixed the UDP socket leaking issue of the DNS module.

Plugin

- ldap-auth-advanced:
  - fix some cache-related issues which cause groups_required to not work properly and unexpected return codes after a non-200 response
- rate-limiting-advanced Fixed an issue where RLA and other similar plugins using the `rate-limiting` library, when used together, would interfere with each other and thus fail to synchronize counter data to the central data store
- rate-limiting-advanced Falling back to local strategy if sync_rate = 0 when redis goes down

Dependency

Core

- Bumped OpenSSL from 3.1.4 to 3.1.5.
- Bump `kong-lua-resty-kafka` to `0.18`.
- Bumped lua-kong-nginx-module to 0.2.3
- Bump `lua-resty-luasocket` to `1.1.2` for fixing luasocket#427.

Performance

Configuration

- Bumped default values of `nginx_http_keepalive_requests` and `upstream_keepalive_max_requests` to `10000`.

2.8.4.7

Release date 2024/02/08

F

P

ncing timers when being executed instead of being created to reduce some meaningless error logs

ous errors are missing when use redis cluster mode connection fails

edis pipeline

changed from a value greater than `0` to `0`, the namespace will be cleared unexpectedly

e the counter syncing timer can't be created or destroyed properly

2

R

F

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- fix ldoc intermittent failure caused by LuaJIT error.
- Fix a bug that GCP backend vault hides the error message when secrets cannot be fetched
- Fix an issue that GCP vault could not fetch secret due to SSL verification failure in CLI mode. Users who use secrets management based on GCP should also ensure `system` CA store is included in `lua_ssl_trusted_certificate` configuration.

Configuration

- respect custom `proxy_access_log`

Plugin

- openid-connect:
  - update time when calculating token expire

Dependency

Core

- Bump resty-openssl from 0.8.25 to 1.0.2
- Bump Alpine base image from `3.16` to `3.19`.
- bump lua-resty-healthcheck to 1.5.4

Performance

Configuration

- Bump `dns_stale_ttl` default to 1 hour so stale DNS record can be used for longer time in case of resolver downtime.

2.8.4.5

Release date 2023/11/28

Feature

Core

- Support observing the time consumed by some components in the given request.
- A unique Request ID is now populated in the error log, access log, error templates, log serializer, and in a new X-Kong-Request-Id header (configurable for upstream/downstream using the `headers` and `headers_upstream` configuration options).

Plugin

- openid-connect:
  - New field `unauthorized_destroy_session`, which when set to true, we destory the session (delete the user's session cookie) when the request is unauthorized. Default to true. Set to false to preserve the session.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Ensure pluginserver protobut includes are placed in the correct path in packages
- consumer-groups: add missing support for tags

PDK

- Fix a bug related to data interference between requests in the kong.log.serialize function.

Plugin

- **collector**:
  - Fix an issue where Kong cannot start after upgrading to version >= 2.8.4.1 due to the deprecated collector plugin still being used.
- openid-connect:
  - Fix issue on token revocation on logout where the code was revoking refresh token when it was supposed to revoke access token when using the discovered revocation endpoint.
- request-validator:
  - Fix an issue where the `allowed_content_types` configuration is unable to contain the "-" character caused by the over-strict validation rule.
- mtls-auth should not cache the network failure when doing revocation check

Admin API

- Fix an issue where unique violation errors were reported while trying to update the user_token with the same value on the same RBAC user.

Dependency

Core

- Bumped lua-resty-aws from 1.3.2 to 1.3.5
- Bump lua-kong-nginx-module from 0.2.0 to 0.2.2
- bump jq to 1.7
- bump OpenSSL to 3.1.4

Kong Manager

- Upgrade moment.js to v2.29.4 to fix a known CVE vulnerability.

Performance

Plugin

- reduce upstream health iteration latency spike during scrape

2.8.4.4

Release date 2023/10/12

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

acks early (CVE-2023-44487)

L

ux 2023 according to AWS's decision

Feature

Plugin

- aws-lambda:
  - the AWS-Lambda plugin has been refactored by using lua-resty-aws as an underlying AWS library. The refactor simplifies the AWS-Lambda plugin code base and adding support for multiple IAM authenticating scenarios.

Bugfix

Core

- Fix a bug that the anonymous report can't be silenced by setting `anonymous_reports=false`.
- The Redis strategy of Rate Limiting catches strategy connection failure.
- Fix an issue can't create developer with non-default workspace.

Plugin

- openid-connect:
  - If an invalid opaque token is provided but verified failed, print the correct error.
- rate-limiting-advanced:
  - cp should not create namespace or do sync.
  - Fix a bug where the rl cluster_events broadcast the wrong data in traditional cluster mode.
- response-transformer-advanced:
  - Do not load response body when `if_status` does not match.

PDK

- Fix a bug in the exit hook that cause customize headers to be lost.

Dependency

Core

- Bumped lua-protobuf from 0.3.3 to 0.4.2
- Bumped lua-resty-aws from 1.0.0 to 1.3.1
- Bumped lua-resty-gcp from 0.0.5 to 0.0.13

2.8.4.2

Release date 2023/07/07

Bugfix

C

ed by the `error_page` directive, could interfere with worker process handling the request when *buffered*

P

here a request with JSON that is not a table failed.:

rver caused a deadlock.:

K

ddition of `static_tags` through the Kong Manager UI.

g correctly.

D

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Bumped `OpenSSL` from 1.1.1t to 3.1.1
- Bumped `lodash` for Dev Portal from 4.17.11 to 4.17.21
- Bumped `lodash` for Kong Manager from 4.17.15 to 4.17.21

2.8.4.1

Release date 2023/05/25

Feature

Core

- Redis Cluster: Added username and password authentication to Redis Cluster 6 and later versions.

Bugfix

Plugin

- **Forward Proxy**:
  - Fixed an issue which occurred when receiving an HTTP `408` from the upstream through a forward proxy. Nginx exited the process with this code, which resulted in Nginx ending the request without any contents.
- request-validator:
  - The plugin now allows requests carrying a `content-type` with a parameter to match its `content-type` without a parameter.

Core

- Fixed an issue where the RBAC token was not re-hashed after an update on the `user_token` field.
- Fixed the Dynatrace implementation. Due to a build system issue, Kong Gateway 2.8.4 packages prior to 2.8.4.1 didn't contain the debug symbols that Dynatrace requires.

Dependency

Core

- Bumped `pgmoon` from 2.2.0.1 to 2.3.2.0.

2.8.4.0

Release date 2023/03/28

Feature

Plugin

- aws-lambda:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`ocol_version`, which lets you select the IMDS protocol version. This option defaults to `v1` and can be set

cluded in the Amazon Linux 2 package.

n a data plane would cause the data plane to stop working after a restart.

`kong.service` in 2.8.1.x and later versions. It has been renamed back to `kong-enterprise-`

8.2.x and later versions.

nen workers were respawned. #9396#9485

grow without bounds if their consumers fail to process the entries. Instead, old batches are now dropped

Plugin

- **mTLS Authentication**:
  - Fixed an issue where the plugin used the old route caches after routes were updated.
- http-log:
  - Fixed an issue in this plugin's batch queue processing, where metrics would be published multiple times. This caused a memory leak, where memory usage would grow without limit.#10052 #10044
- key-auth-enc:
  - Fixed an issue where using an API key that exists in multiple workspaces caused a 401 error. This occurred because of a caching issue.
- Added the missing `protocols` field to the following plugin schemas: Azure Functions (`azure-functions`)
- Added the missing `protocols` field to the following plugin schemas: gRPC Gateway (`grpc-gateway`)
- Added the missing `protocols` field to the following plugin schemas: gRPC Web (`grpc-web`)
- Added the missing `protocols` field to the following plugin schemas: Serverless pre-function (`pre-function`)
- Added the missing `protocols` field to the following plugin schemas: Prometheus (`prometheus`)
- Added the missing `protocols` field to the following plugin schemas: Proxy Caching (`proxy-cache`)
- Added the missing `protocols` field to the following plugin schemas: Request Transformer (`request-transformer`)
- Added the missing `protocols` field to the following plugin schemas: Session (`session`)
- Added the missing `protocols` field to the following plugin schemas: Zipkin (`zipkin`)

2.8.2.4

Release date 2023/01/23

Bugfix

Core

- Kong Gateway now statically links the BoringSSL PCRE library. This fixes an issue introduced in 2.8.2.3, where the BoringSSL library was dynamically linked, causing regex compilation to fail when routing requests with some versions of the library.

2.8.2.3

Release date 2023/01/06

Bugfix

Core

- Fixed a role precedence issue with RBAC. RBAC rules involving deny (negative) rules now correctly take precedence over allow (non-negative) roles.
- Fixed workspace filtering pagination on the overview page.
- Fixed a router issue where, in an environment with more than 50,000 routes, attempting to update a route caused a `500` error response.

ssaging protocol connection broke in hybrid mode.

SSL was configured on PostgreSQL, and the Kong Gateway had `stream_listen` configured with a stream

P

error caused by empty headers. This error occurred when using this plugin with the Kong Ingress

a string value).

ally forward an unverified token to the upstream.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed UUID pattern matching.

- prometheus:
  - Provided options to reduce the plugin's impact on performance. Added new `kong.conf` options to switch high cardinality metrics `on` or `off`: `prometheus_plugin_status_code_metrics`, `prometheus_plugin_latency_metrics`, `prometheus_plugin_bandwidth_metrics`, and `prometheus_plugin_upstream_health_metrics`.

- rate-limiting-advanced:
  - Fixed a maintenance cycle lock leak in the `kong_locks` dictionary. Kong Gateway now clears old namespaces from the maintenance cycle schedule when a namespace is updated.

- request-transformer:
  - Fixed an issue where empty arrays were being converted to empty objects. Empty arrays are now preserved.

Known Issues

Core

- A required PCRE library is dynamically linked, where prior versions statically linked the library. Depending on the system PCRE version, this may cause regex compilation to fail when routing requests. Starting in 2.8.2.4 and later, Kong Gateway will return to statically linking the PCRE library.

2.8.2.2

Release date 2022/12/01

Bugfix

Core

- Added batch queues for the Datadog and StatsD plugins to reduce timer usage, fixing a `lua_max_running_timers are not enough` timer error.

  Whenever a request was processed, a new running timer was instantly created during the log phase. This was causing a shortage of timers under heavy traffic and led to unpredictable consequences, where internal timers were killed randomly and couldn't recover automatically. This would then trigger a `lua_max_running_timers are not enough` timer error and cause data planes to crash.

  #9521

- Fixed a timer leak that occurred whenever the generic messaging protocol connection would break in hybrid mode.

2.8.2.1

Release date 2022/11/21

Bugfix

Core

- **Kong Manager:**
  - Fixed an issue where admins needed the specific `rbac/role` permission to edit RBAC roles. Now, admins can edit RBAC roles with the `/admins` permission.
  - Fixed an issue where the client certificate ID didn't display properly in the upstream update form.
  - Fixed an issue in the service documents UI which allowed users to upload multiple documents. Since each service only supports one document, the documents would not display correctly. Uploading a new document now overrides the previous document.

n the global workspace dashboard wasn't clickable on the first page load.
eleted roles.
sly, `VUE_APP_NEW_RELIC_LICENSE_KEY` and`VUE_APP_SEGMENT_WRITE_KEY` were being exposed in Kong

specific endpoints (for example, an individual service or route) were not reflected in the Kong Manager

didn't support group names with spaces.
ds, where right-clicking on **View All** and choosing "Open Link in New Tab" or "Copy Link" for services,
ace and caused an `HTTP 404` error.

xamples weren't rendered when media type was vendor-specific.

h checks, and checking their status via the `upstreams/<upstream>/health` endpoint results in the status
m>/health endpoint was incorrectly reporting targets with `weight=0` as `HEALTHY`, and the health check was

permissions to access logs.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Fixed an issue where operational attributes referenced by `group_member_attribute` weren't returned in search query results.

- **mTLS Authentication**:
  - Fixed an issue where the plugin was causing requests to silently fail on Kong Gateway data planes.
- azure-functions:
  - Fixed an issue where calls made by this plugin would fail in the following situations:
    - The plugin was associated with a route that had no service.
    - The route's associated service had a `path` value.
  - The plugin was associated with a route that had no service.
  - The route's associated service had a `path` value.
- kafka-upstream:
  - Fixed the `Bad Gateway` error that would occur when using the Kafka Upstream plugin with the configuration `producer_async=false`.
- request-transformer:
  - Fixed an issue where empty arrays were being converted to empty objects. Empty arrays are now preserved.
- response-transformer:
  - Fixed an issue where the plugin couldn't process string responses.

2.8.2.0

Release date 2022/10/12

Bugfix

Core

- **Kong Manager** :
  - Fixed an issue where workspaces with zero roles were not correctly sorted by the number of roles.
  - Fixed the Cross Site Scripting (XSS) security vulnerability in the Kong Manager UI.
  - Fixed an issue where registering an admin without `admin_gui_auth` set resulted in a `500` error.
  - Fixed an issue that allowed unauthorized IDP users to log in to Kong Manager. These users had no access to any resources in Kong Manager, but were able to go beyond the login screen.
- Fixed OpenSSL vulnerabilities CVE-2022-2097 and CVE-2022-2068.
- Hybrid mode: Fixed an issue with consumer groups, where the control plane wasn't sending the correct number of consumer entries to data planes.
- Hybrid mode: Fixed an issue where sending a `PATCH` request to update a route after restarting a control plane caused a 500 error response.

Plugin

- **Forward Proxy**:
  - If the `https_proxy` configuration parameter is not set, it now defaults to `http_proxy` to avoid DNS errors.
- **GraphQL Proxy Cache Advanced**:

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

access phase (only in: log), which was preventing the plugin from working consistently.

onment variables in the ECS environment, causing permission errors.

cluster strategy in hybrid or DB-less modes instead of crashing.

tributes.

s couldn't be added to more than one workspace, and permissions were not being updated.

verted to empty objects. Empty arrays are now preserved.

- Fixed an issue where URIs that included `%20` or a whitespace would return a `400 Bad Request`.

2.8.1.4

Release date 2022/08/23

Dependency

Core

- Bump `lua-resty-aws` version to 0.5.4 to reduce memory usage when AWS vault is enabled. #23
- Bump `lua-resty-gcp` version to 0.0.5 to reduce memory usage when GCP vault is enabled. #7

2.8.1.3

Release date 2022/08/05

Feature

Core

- Added GCP integration support for the secrets manager. GCP is now available as a vault backend.

Plugin

- aws-lambda:
  - Added support for cross-account invocation through the `aws_assume_role_arn` and`aws_role_session_name` configuration parameters.#8900

Bugfix

Core

- Fixed an issue with excessive log file disk utilization on control planes.
- Fixed an issue with keyring encryption, where keyring was not decrypting keys after a soft reload.
- The router now detects static route collisions inside the current workspace, as well as with other workspaces.
- When using a custom plugin in a hybrid mode deployment, the control plane now detects compatibility issues and stops sending the plugin configuration to data planes that can't use it. The control plane continues sending the custom plugin configuration to compatible data planes.
- Optimized the Kong PDK function `kong.response.get_source()`.

Plugin

- acme:
  - The `domains` plugin parameter can now be left empty. When `domains` is empty, all TLDs are allowed. Previously, the parameter was labelled as optional, but leaving it empty meant that the plugin retrieved no certificates at all.
- mocking:
  - Fixed an issue where the plugin didn't accept empty values in examples.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ecision in Cassandra.

admin was created with no roles, the admin would have access to the first workspace listed alphabetically.

list was empty after sorting by the SSL certificate ID field. In 2.8.1.1, the SSL certificate ID field in the

Bugfix

Core

- Fixed an issue in hybrid mode where, if a service was set to `enabled: false` and that service had a route with an enabled plugin, any new data planes would receive empty configuration.
- Fixed a timer leak that occurred when `worker_consistency` was set to `eventual` in `kong.conf`. This issue caused timers to be exhausted and failed to start any other timers used by Kong Gateway, resulting in a `too many pending timers` error.
- Fixed memory leaks coming from `lua-resty-lock`.
- Fixed global plugins can operate out of the workspace scope

kong-manager-and-dev-portal

- Fixed an issue where Kong Manager did not display all Dev Portal developers in the organization.
- Fixed an issue that prevented developer role assignments from displaying in Kong Manager. When viewing a role under the Permissions tab in the Dev Portal section, the list of developers wouldn't update when a new developer was added. Kong Manager was constructing the wrong URL when retrieving Dev Portal assignees.
- Fixed empty string handling in Kong Manager. Previously, Kong Manager was handling empty strings as `""` instead of a null value.
- Improved Kong Manager styling by fixing an issue where content didn't fit on object detail pages.
- Fixed an issue that sometimes prevented clicking Kong Manager links and buttons in Safari.
- Fixed an issue where users were being navigated to the object detail page after clicking on the "Copy ID" button from the object list.
- Fixed an issue where the number of requests and error rate were not correctly displaying when Vitals was disabled.

Plugin

- http-log:
  - Log output is now restricted to the workspace the plugin is running in. Previously, the plugin could log requests from outside of its workspace.
- mocking:
  - Fixed an issue where `204` responses were not handled correctly and you would see the following error:`"No examples exist in API specification for this resource"`.
  - `204` response specs now support empty content elements.
- rate-limiting:
  - Fixed a PostgreSQL deadlock issue that occurred when the `cluster` policy was used with two or more metrics (for example, `second` and `day`.)

Deprecated

Core

- **Amazon Linux 1** : Support for running Kong Gateway on Amazon Linux 1 is now deprecated, as theAmazon Linux (1) AMI has ended standard support as of December 31, 2020. Starting with Kong Gateway 3.0.0.0, Kong is not building new Amazon Linux 1 images or packages, and Kong will not test package installation on Amazon Linux 1.

  If you need to install Kong Gateway on Amazon Linux 1, see the documentation for previous versions.

  8 ("Jessie") is now deprecated, as Debian 8 ("Jessie") has reached End of Life (EOL). Starting with Kong
  ssie") images or packages, and Kong will not test package installation on Debian 8 ("Jessie").

  "), see the documentation for previous versions.

  buntu 16.04 ("Xenial") is now deprecated, as Standard Support for Ubuntu 16.04 has ended as of April,
  uilding new Ubuntu 16.04 images or packages, and Kong will not test package installation on Ubuntu

  e the documentation for previous versions.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- You can now enable application status and application request emails for the Developer Portal using the following configuration parameters:
  - `portal_application_status_email`: Enable to send application request status update emails to developers.
  - `portal_application_request_email`: Enable to send service access request emails to users specified in `smtp_admin_emails`.
  - `portal_smtp_admin_emails`: Specify the email addresses to send portal admin emails to, overriding values set in `smtp_admin_emails`.
- Added the ability to use `email.developer_meta` fields in portal email templates. For example, `{{email.developer_meta.preferred_name}}`.

Plugin

- **mTLS Authentication**:
  - Introduced certificate revocation list (CRL) and OCSP server support with the following parameters: `http_proxy_host`, `http_proxy_port`, `https_proxy_host`, and `https_proxy_port`.
- aws-lambda:
  - When working in proxy integration mode, the `statusCode` field now accepts string datatypes.
- kafka-upstream:
  - Added support for the `SCRAM-SHA-512` authentication mechanism.

Bugfix

Core

- Improved Kong Admin API and Kong Manager performance for organizations with many entities.
- Fixed an issue with keyring encryption, where the control plane would crash if any errors occurred during the initialization of the keyring module.
- Fixed an issue where Kong Manager did not display all RBAC users and Consumers in the organization.
- Fixed an issue where some areas in a row of a list were not clickable.

Plugin

- **Forward Proxy**:
  - Fixed an `invalid header value` error for HTTPS requests. The plugin now accepts multi-value response headers.
  - Fixed an error where basic authentication headers containing the =character weren't forwarded.
  - Fixed request errors that occurred when a scheme had no proxy set. The`https` proxy now falls back to the `http` proxy if not specified, and the`http` proxy falls back to `https`.
- graphql-rate-limiting-advanced:
  - Fixed `deserialize_parse_tree` logic when building GraphQL AST with non-nullable or list types.
- rate-limiting-advanced:
  - Fixed rate limiting advanced errors that appeared when the Rate Limiting Advanced plugin was not in use.
  - Fixed an error where rate limiting counters were not updating response headers due to incorrect key expiration tracking. Redis key expiration is now tracked properly in `lua_shared_dict kong_rate_limiting_counters`.

2.8.1.0

R

F

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

space=* would not let the /kong endpoint be accessed from all workspaces

vel endpoint=* permission could not add any RBAC rules, even if they had endpoint=/rbac permissions.
workspace only.

s a comma in the provided value
wing the plugin configuration
404 error
default workspaces could not see any workspaces
access to non-default workspaces
ng Cassandra
AC edit page

    - Performing an action on an upstream target no longer leads to a 404 error

- Developer Portal
    - Information about the current session is now bound to an nginx worker thread. This prevents data leaks when a worker is handling multiple requests at the same time: Information about the current session is now bound to an nginx worker thread. This prevents data leaks when a worker is handling multiple requests at the same time
- Keys are no longer rotated unexpectedly when a node restarts
- Add cache when performing RBAC token verification
- The log message “plugins iterator was changed while rebuilding it” was incorrectly logged as an `error`. This release converts it to the `info` log level.
- Fixed a 500 error when rate limiting counters are full with the Rate Limiting Advanced plugin
- Improved the performance of the router, plugins iterator and balancer by adding conditional rebuilding

Plugin

- **Forward Proxy**:
    - Use lowercase when overwriting the `host` header
- **LDAP Auth Advanced**:
    - Support passwords that contain a `:` character
- http-log:
    - Include provided query string parameters when sending logs to the `http_endpoint`
- jwt-signer:
    - Implement the `enable_hs_signatures` option to enable JWTs signed with HMAC algorithms
- openid-connect:
    - Provide valid upstream headers e.g. `X-Consumer-Id`, `X-Consumer-Username`
- rate-limiting-advanced:
    - Skip namespace creation if the plugin is not enabled. This prevents the error “[rate-limiting-advanced] no shared dictionary was specified” being logged.
- StatsD Advanced: Added support for setting `workspace_identifier` to `workspace_name`

Dependency

Core

- Bumped `openssl` from 1.1.1k to 1.1.1n to resolve CVE-2022-0778 #8635
- Bumped `openresty` from 1.19.3.2 to 1.19.9.1 #7727

2.8.0.0

Release date 2022/03/02

Feature

C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

ted instances only)

sing the old **View** icon.
wn, which is located above the table.

onadmin_auto_create_rbac_token_disabledto enable or disable RBAC tokens when automatically

`api` signal foranonymous_reports.

anagement and vault support. You can now store confidential values such as usernames and passwords as
e these secrets, making your environment more secure.

lementations:

- HashiCorp Vault

    - Environment variable

  As part of this support, some plugins have certain fields marked as_referenceable_. See the plugin section of the Kong Gateway 2.8 changelog for details.

  Test out secrets management using thegetting started guide, and check out the documentation for the Kong Admin API `/vaults-beta` entity.

  > This feature is in beta. It has limited support and implementation details may change. This means it is intended for testing in staging environments only, and **should not** be deployed in production environments.

- You can customize the transparent dynamic TLS SNI name.

  Thanks, @Murphy-hub! #8196

- Routes now support matching headers with regular expressions.

  Thanks, @vanhtuan0409! #6079

- You can now configure `cluster_max_payload`for hybrid mode deployments. This configuration option sets the maximum payload size allowed to be sent across from the control plane to the data plane. If your environment has large configurations that generate `payload too big` errors and don't get applied to the data planes, use this setting to adjust the limit.

  Thanks, @andrewgkew! #8337

dev-portal

- The new `/developers/export` endpoint lets you export the list of developers and their statuses into CSV format.

Performance

- Improved the calculation of declarative configuration hash for big configurations. The new method is faster and uses less memory.#8204
- Multiple improvements in the Router, including:
  - The router builds twice as fast
  - Failures are cached and discarded faster (negative caching)
  - Routes with header matching are cached

  These changes should be particularly noticeable when rebuilding in DB-less environments.#8087#8010

Admin API

- The current declarative configuration hash is now returned by the `status`endpoint when Kong node is running in DB-less or data plane mode.#8214#8425

Plugin

- **Proxy Cache Advanced**:
  - Added Redis ACL support (Redis v6.0.0+ and Redis Sentinel v6.2.0+).
  - Added the `redis.sentinel_username` and `redis.sentinel_password` configuration parameters.
  - **Beta feature:** The `redis.password`, `redis.sentinel_username`, and `redis.sentinel_password`configuration fields are now marked as referenceable, which means they can be securely stored as secretsin a vault. References must follow a specific format.
- acme:
  - Added the `rsa_key_size` configuration parameter.

    Thanks, lodrantl! #8114

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

`_name`. This parameter accepts a header name that, when present on a request, overrides the configured

alue `always`, the request will always go to the canary upstream.
the request will never go to the canary upstream.

`lways`, the request will always go to the canary upstream.

quest will never go to the canary upstream.

`s_proxy_host`, and`https_proxy_port` configuration parameters for mTLS support.

d `proxy_host` fields, which are now **deprecated** and planned to be removed in 3.x.x.

tion fields are now marked as referenceable, which means they can be securely stored assecretsin a vault.

- graphql-rate-limiting-advanced:

- Added Redis ACL support (Redis v6.0.0+ and Redis Sentinel v6.2.0+).
- Added the `redis.username` and `redis.sentinel_username` configuration parameters.
- **Beta feature:** The `redis.username`, `redis.password`, `redis.sentinel_username`, and `redis.sentinel_password`configuration fields are now marked as referenceable, which means they can be securely stored assecretsin a vault. References must follow a specific format.

- jq:
  - Use response buffering from the PDK.
- kafka-upstream:
  - Added the ability to identify a Kafka cluster using the `cluster_name` configuration parameter. By default, this field generates a random string. You can also set your own custom cluster identifier.
  - **Beta feature:** The `authentication.user` and `authentication.password` configuration fields are now marked as referenceable, which means they can be securely stored assecretsin a vault. References must follow a specific format.
- ldap-auth-advanced:
  - **Beta feature:** The `ldap_password` and `bind_dn` configuration fields are now marked as referenceable, which means they can be securely stored assecretsin a vault. References must follow a specific format.
- openid-connect:
  - Added Redis ACL support (Redis v6.0.0+) for storing and retrieving a session. Use the `session_redis_username` and `session_redis_password` configuration parameters to configure it.

    These parameters replace the `session_redis_auth` field, which is now **deprecated** and planned to be removed in 3.x.x.
  - Added support for distributed claims. Set the `resolve_distributed_claims` configuration parameter to `true` to tell OIDC to explicitly resolve distributed claims.

    Distributed claims are represented by the `_claim_names` and `_claim_sources` members of the JSON object containing the claims.
  - **Beta feature:** The `client_id`, `client_secret`, `session_secret`, `session_redis_username`, and `session_redis_password` configuration fields are now marked as referenceable, which means they can be securely stored assecretsin a vault. References must follow a specific format.
- prometheus:
  - Added three new metrics:
    - `kong_db_entities_total` (gauge): total number of entities in the database.
    - `kong_db_entity_count_errors` (counter): measures the number of errors encountered during the measurement of `kong_db_entities_total`.
    - `kong_nginx_timers` (gauge): total number of Nginx timers, in Running or Pending state. Tracks `ngx.timer.running_count()` and`ngx.timer.pending_count()`.#8387
  - `kong_db_entities_total` (gauge): total number of entities in the database.
  - `kong_db_entity_count_errors` (counter): measures the number of errors encountered during the measurement of `kong_db_entities_total`.
  - `kong_nginx_timers` (gauge): total number of Nginx timers, in Running or Pending state. Tracks `ngx.timer.running_count()` and`ngx.timer.pending_count()`.#8387
- rate-limiting-advanced:
  - Added Redis ACL support (Redis v6.0.0+ and Redis Sentinel v6.2.0+).
  - Added the `redis.username` and `redis.sentinel_username` configuration parameters.
  - **Beta feature:** The `redis.username`, `redis.password`, `redis.sentinel_username`, and `redis.sentinel_password`configuration fields are now marked as referenceable, which means they can be securely stored assecretsin a vault. References must follow a specific format.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

s Sentinel v6.2.0+).

er.

213

l is now marked as referenceable, which means it can be securely stored as asecretin a vault. References

Bugfix

Core

- Fixed a timer leak that caused the timers to be exhausted and failed to start any other timers used by Kong, showing the error `too many pending timers`.
- Fixed an issue where, if `data_plane_config_cache_mode` was set to `off`, the data plane received no updates from the control plane.
- Fixed `attempt to index local 'workspace'` error, which occurred when accessing Routes or Services using TLS.
- Fixed an issue where `cluster_telemetry_server_name`was not automatically generated and registered if it was not explicitly set.
- Fixed the `cluster_allowed_common_names`setting. When using PKI for certificate verification in hybrid mode, you can now configure a list of Common Names allowed to connect to a control plane with the option. If not set, only data planes with the same parent domain as the control plane cert are allowed.
- When the Router encounters an SNI FQDN with a trailing dot (`.`), the dot will be ignored, since according toRFC-3546the dot is not part of the hostname.#8269
- Fixed a bug in the Router that would not prioritize the routes with both a wildcard and a port (`route.*:80`) over wildcard-only routes (`route.*`), which have less specificity.#8233
- The internal DNS client isn't confused by the single-dot (`.`) domain, which can appear in `/etc/resolv.conf` in special cases like `search .`.#8307
- The Cassandra connector now records migration consistency level.

  Thanks, @mpenick! #8226

dev-portal

- When the SMTP configuration was broken or unresponsive, the API would respond with an error message that was a JavaScript Object (`[Object object]`) instead of a string. This happened when a user was registering on any given portal with broken SMTP. Now, if there is an error, the API responds with the string`Error sending email`.
- The `/document_objects` and `/services/:id/document_objects` endpoints no longer accept multiple documents per service. This was an issue, as each service can only have one document. Instead, posting a document to one of these endpoints now overrides the previous document.

balancer

- Targets now keep their health status when upstreams are updated.#8394
- One debug message which was erroneously using the `error` log level has been downgraded to the appropriate `debug` log level.#8410

Clustering

- Replaced a cryptic error message with a more useful one when there is a failure on SSL when connecting with the control plane.#8260

Admin API

- Fixed an incorrect `next` field that appeared when paginating Upstreams.#8249

PDK

- Phase names are now correctly selected when performing phase checks.#8208
- Fixed a bug in the go-PDK where, if `kong.request.getrawbody` was big enough to be buffered into a temporary file, it would return an an empty string.#8390

Plugin

tructure and representation of null values, which provoked an error on init with the go-pdk.#8267

would attempt to load HTTP headers.#8414

behind a reverse proxy.

when caching large files.

or, which occurred when accessing Routes or Services using TLS.

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- aws-lambda:
  - Fixed incorrect behavior when configured to use an HTTP proxy and deprecated the `proxy_scheme` config attribute for removal in 3.0.#8406
- cors:
  - The CORS plugin does not send the `Vary: Origin` header anymore when the header `Access-Control-Allow-Origin` is set to `*`.

    Thanks, @jkla-dr! #8401
- datadog:
  - The plugin schema now lists the default values for configuration options in a single place instead of in two separate places.#8315
- exit-transformer:
  - Fixed an issue where the Exit Transformer plugin would break the plugin iterator, causing later plugins not to run.
- jq:
  - If plugin has no output, it will now return the raw body instead of attempting to restore the original response body.
- jwt-signer:
  - Fixed an issue where the `enable_hs_signatures` configuration parameter did not work. The plugin now defines expiry earlier to avoid arithmetic on a nil value.
- oauth2:
  - The plugin clears the `X-Authenticated-UserId` and`X-Authenticated-Scope` headers when it is configured in logical OR and is used in conjunction with another authentication plugin.#8422
- openid-connect:
  - Fixed negative caching, which was loading wrong a configuration value.
- rate-limiting-advanced:
  - Fixed a 500 error that occurred when consumer groups were enforced but no proper configurations were provided. Now, if no specific consumer group configuration exists, the consumer group defaults to the original plugin configuration.
  - Fixed a timer leak that caused the timers to be exhausted and failed to start any other timers used by Kong, showing the error `too many pending timers`.

    Before, the plugin used one timer for each namespace maintenance process, increasing timer usage on instances with a large number of rate limiting namespaces. Now, it uses a single timer for all namespace maintenance.
  - Fixed an issue where the `local` strategy was not working with DB-less and hybrid deployments. We now allow `sync_rate = null` and `sync_rate = -1`when a `local` strategy is defined.
- rate-limiting:
  - Fixed a 500 error associated with performing arithmetic functions on a nil value by adding a nil value check after performing `ngx.shared.dict` operations.
- response-transformer-advanced:
  - In the `body_filter` phase, the plugin now sets the body to an empty string instead of `nil`.

Kong Manager

- Fixed an issue where OIDC authentication into Kong Manager failed when used with Azure AD.
- Fixed a performance issue with the Teams page in Kong Manager.
- the checkbox for the OAuth2 plugin's `hash_secret` value was labelled as *Required* and users were not
- ugin configuration when attempting to clear the `service.id` from a plugin.
- re a new route would default to `http` as the supported protocol. Now, creating a Route picks up the correct
- on for Route and Service objects on their configuration pages.
- which caused the error"`attempt to call method 'select_by_username_ignore_case' (a nil`
- tokens are no longer tracked by the workspace entity counter, as the count is not needed by the Kong
- ed by the **Applied To** column.

D

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

Core

- Bumped OpenSSL from 1.1.1l to 1.1.1m#8191
- Bumped `resty.session` from 3.8 to 3.10#8294
- Bumped `lua-resty-openssl` to 0.8.5#8368
- Bumped `lodash` for Dev Portal from 4.17.11 to 4.17.21
- Bumped `lodash` for Kong Manager from 4.17.15 to 4.17.21

Deprecated

Core

- The external `go-pluginserver` project is considered deprecated in favor of the embedded server approach.
- Starting with Kong Gateway 2.8.0.0, Kong is not building new open-source CentOS images. Support for running open-source Kong Gateway on CentOS on is now deprecated, as CentOS has reached End of Life (OEL).

  Running Kong Gateway Enterprise on CentOS is currently supported, but CentOS is planned to be fully deprecated in Kong Gateway 3.x.x.
- OpenID Connect plugin: The `session_redis_auth` field is now deprecated and planned to be removed in 3.x.x. Use`session_redis_username` and `session_redis_password` instead.
- Forward Proxy Advanced plugin: The `proxy_port` and `proxy_host` fields are now deprecated and planned to be removed in 3.x.x. Use`http_proxy_host` and `http_proxy_port`, or `https_proxy_host` and`https_proxy_port` instead.
- AWS Lambda plugin: The `proxy_scheme` field is now deprecated and planned to be removed in 3.x.x.

Related Documentation
Gateway Documentation
Tags
#changelog
Related Resources
Kong Gateway breaking changes

- Kong Gateway changelog
- 3.16.0.0
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.15.0.6
  - Bugfix
  - Dependency
  - Performance
- 3.15.0.5
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.15.0.4
  - Feature
  - Bugfix
- 3.15.0.3
  - Bugfix

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- 3.14.0.12

  - Bugfix
  - Dependency
- 3.14.0.11
  - Feature
  - Bugfix
- 3.14.0.10
  - Feature
  - Bugfix
  - Performance
- 3.14.0.9
  - Bugfix
  - Dependency
- 3.14.0.8
  - Bugfix
- 3.14.0.7
  - Bugfix
  - Dependency
  - Performance
- 3.14.0.6
  - Bugfix
- 3.14.0.5
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.14.0.4
  - Bugfix
  - Dependency
- 3.14.0.3
  - Bugfix
- 3.14.0.2
  - Bugfix
  - Dependency
- 3.14.0.1
  - Bugfix
- 3.14.0.0
  - Breaking Change
  - Deprecation
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.13.0.10
  - Bugfix
  - Performance
- 3.13.0.9
  - Bugfix
  - Dependency
- 3.13.0.8
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.13.0.7
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.13.0.6
  - Bugfix

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- 3.12.0.12
  - Bugfix
- 3.12.0.11
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.12.0.10
  - Bugfix
  - Dependency
- 3.12.0.9
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.12.0.8
  - Bugfix
- 3.12.0.7
  - Bugfix
  - Dependency
- 3.12.0.6
  - Bugfix
  - Dependency
- 3.12.0.5
  - Bugfix
  - Dependency
  - Performance
- 3.12.0.4
  - Bugfix
  - Dependency
- 3.12.0.3
  - Bugfix
- 3.12.0.2
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.12.0.1
  - Bugfix
- 3.12.0.0
  - Breaking Change
  - Deprecation
  - Feature
  - Bugfix
  - Dependency
  - Performance
  - Known Issues
- 3.11.0.14
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.11.0.13
  - Bugfix
- 3.11.0.12
  - Bugfix
  - Dependency
- 3.11.0.11
  - Bugfix
- 3.11.0.10

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Bugfix
  - Dependency
  - Performance
- 3.11.0.2
  - Bugfix
- 3.11.0.1
  - Bugfix
  - Dependency
- 3.11.0.0
  - Deprecation
  - Feature
  - Bugfix
  - Dependency
  - Performance
  - Known Issues
- 3.10.0.18
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.10.0.17
  - Bugfix
  - Dependency
- 3.10.0.16
  - Bugfix
  - Performance
- 3.10.0.15
  - Bugfix
  - Dependency
  - Performance
- 3.10.0.14
  - Bugfix
- 3.10.0.13
  - Bugfix
- 3.10.0.12
  - Bugfix
  - Dependency
- 3.10.0.11
  - Feature
  - Bugfix
  - Dependency
- 3.10.0.10
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.10.0.9
  - Bugfix
  - Dependency
- 3.10.0.8
  - Bugfix
  - Performance
- 3.10.0.7
  - Bugfix
  - Dependency
  - Performance
- 3.10.0.6
  - Feature
  - Bugfix
- 3.10.0.5

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

    - Deprecation
    - Feature
    - Bugfix
    - Dependency
    - Performance
    - Known Issues
- 3.9.1.2
    - Feature
    - Bugfix
    - Dependency
- 3.9.1.1
    - Bugfix
- 3.9.1.0
    - Feature
    - Bugfix
- 3.9.0.1
    - Bugfix
    - Dependency
- 3.9.0.0
    - Deprecation
    - Feature
    - Bugfix
    - Dependency
    - Known Issues
- 3.8.1.2
    - Feature
    - Bugfix
    - Dependency
- 3.8.1.1
    - Deprecation
    - Feature
    - Bugfix
    - Dependency
- 3.8.1.0
    - Feature
    - Bugfix
    - Dependency
- 3.8.0.0
    - Deprecation
    - Feature
    - Bugfix
    - Dependency
    - Performance
    - Feature
    - Known Issues
- 3.7.1.5
    - Feature
    - Bugfix
- 3.7.1.4
    - Feature
    - Bugfix
    - Dependency
- 3.7.1.3
    - Feature
    - Bugfix
    - Dependency
- 3.7.1.2
    - Deprecation
    - Feature
    - Bugfix

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Dependency
- 3.6.1.6
  - Bugfix
- 3.6.1.5
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.6.1.4
  - Feature
  - Bugfix
  - Dependency
- 3.6.1.3
  - Bugfix
- 3.6.1.2
  - Feature
  - Bugfix
  - Dependency
- 3.6.1.1
  - Bugfix
- 3.6.1.0
  - Feature
  - Bugfix
  - Performance
- 3.6.0.0
  - Breaking Change
  - Deprecation
  - Feature
  - Bugfix
  - Dependency
  - Performance
  - Known Issues
- 3.5.0.7
  - Deprecation
  - Feature
- 3.5.0.6
  - Bugfix
- 3.5.0.5
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.5.0.4
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.5.0.3
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
- 3.5.0.2
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.5.0.1

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Performance
- 3.4.3.24
  - Bugfix
  - Dependency
- 3.4.3.23
  - Bugfix
  - Dependency
  - Performance
- 3.4.3.22
  - Bugfix
  - Dependency
  - Performance
- 3.4.3.21
  - Bugfix
  - Dependency
- 3.4.3.20
  - Bugfix
  - Dependency
- 3.4.3.19
  - Bugfix
- 3.4.3.18
  - Feature
  - Bugfix
  - Dependency
- 3.4.3.17
  - Feature
  - Bugfix
  - Dependency
- 3.4.3.16
  - Dependency
- 3.4.3.15
  - Bugfix
  - Dependency
- 3.4.3.14
  - Bugfix
- 3.4.3.13
  - Feature
  - Bugfix
  - Dependency
- 3.4.3.12
  - Feature
  - Bugfix
  - Dependency
- 3.4.3.11
  - Bugfix
- 3.4.3.10
  - Bugfix
  - Dependency
- 3.4.3.9
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.4.3.8
  - Feature
  - Bugfix
  - Dependency
- 3.4.3.7
  - Bugfix
  - Dependency

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- 3.4.3.1
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 3.4.2.0
  - Feature
  - Bugfix
  - Dependency
- 3.4.1.1
  - Bugfix
  - Dependency
- 3.4.1.0
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
- 3.4.0.0
  - Feature
  - Bugfix
  - Dependency
  - Breaking Change
  - Known Issues
- 3.3.1.1
  - Bugfix
  - Dependency
- 3.3.1.0
  - Deprecation
  - Bugfix
- 3.3.0.0
  - Feature
  - Bugfix
  - Dependency
  - Breaking Change
  - Known Issues
- 3.2.2.5
  - Bugfix
  - Dependency
- 3.2.2.4
  - Bugfix
  - Dependency
  - Breaking Change
- 3.2.2.3
  - Deprecation
  - Bugfix
- 3.2.2.2
  - Bugfix
- 3.2.2.1
  - Deprecation
  - Bugfix
- 3.2.2.0
  - Bugfix
  - Known Issues
- 3.2.1.0
  - Deprecation
  - Feature
  - Bugfix
  - Dependency
  - Breaking Change

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

- Breaking Change
- 3.0.1.0
  - Feature
  - Bugfix
- 3.0.0.0
  - Feature
  - Bugfix
  - Dependency
  - Breaking Change
  - Known Issues
- 2.8.4.14
  - Feature
  - Bugfix
  - Dependency
- 2.8.4.13
  - Breaking Change
  - Bugfix
- 2.8.4.12
  - Deprecation
  - Dependency
- 2.8.4.11
  - Bugfix
- 2.8.4.10
  - Feature
  - Bugfix
  - Dependency
- 2.8.4.9
  - Bugfix
- 2.8.4.8
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 2.8.4.7
  - Bugfix
- 2.8.4.6
  - Bugfix
  - Dependency
  - Performance
- 2.8.4.5
  - Feature
  - Bugfix
  - Dependency
  - Performance
- 2.8.4.4
  - Bugfix
- 2.8.4.3
  - Breaking Change
  - Feature
  - Bugfix
  - Dependency
- 2.8.4.2
  - Bugfix
  - Dependency
- 2.8.4.1
  - Feature
  - Bugfix
  - Dependency
- 2.8.4.0
  - Feature

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

  - Feature
  - Bugfix
- 2.8.1.0
  - Bugfix
  - Dependency
- 2.8.0.0
  - Feature
  - Bugfix
  - Dependency
  - Deprecated

Did this doc help?

Yes No

Something wrong?
Report an Issue | Edit this Page
View all Kong documentation

Help us make these docs great!

Kong Developer docs are open source. If you find these useful and want to make them better, contribute today!
Contribute

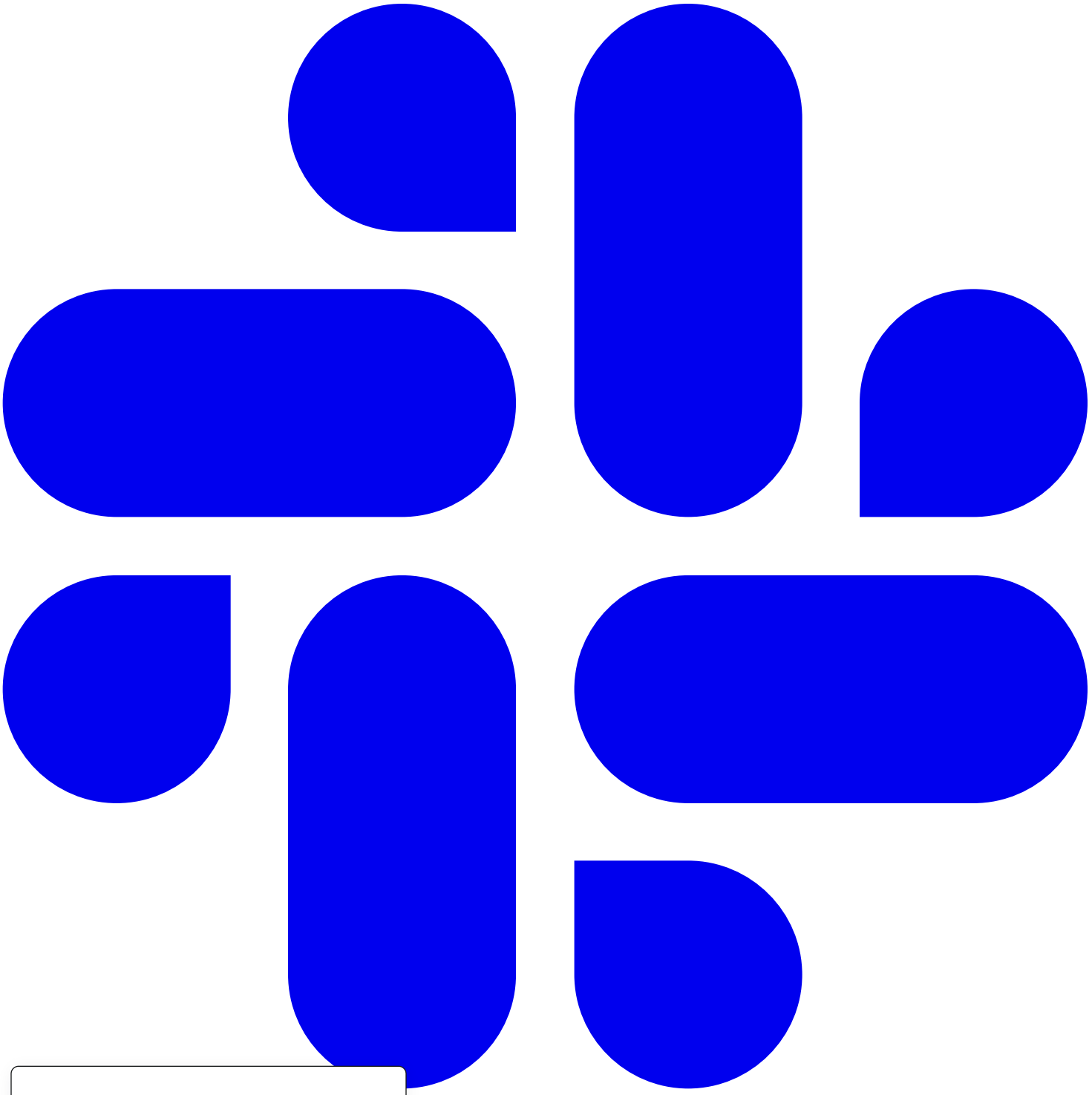
Still need help?

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

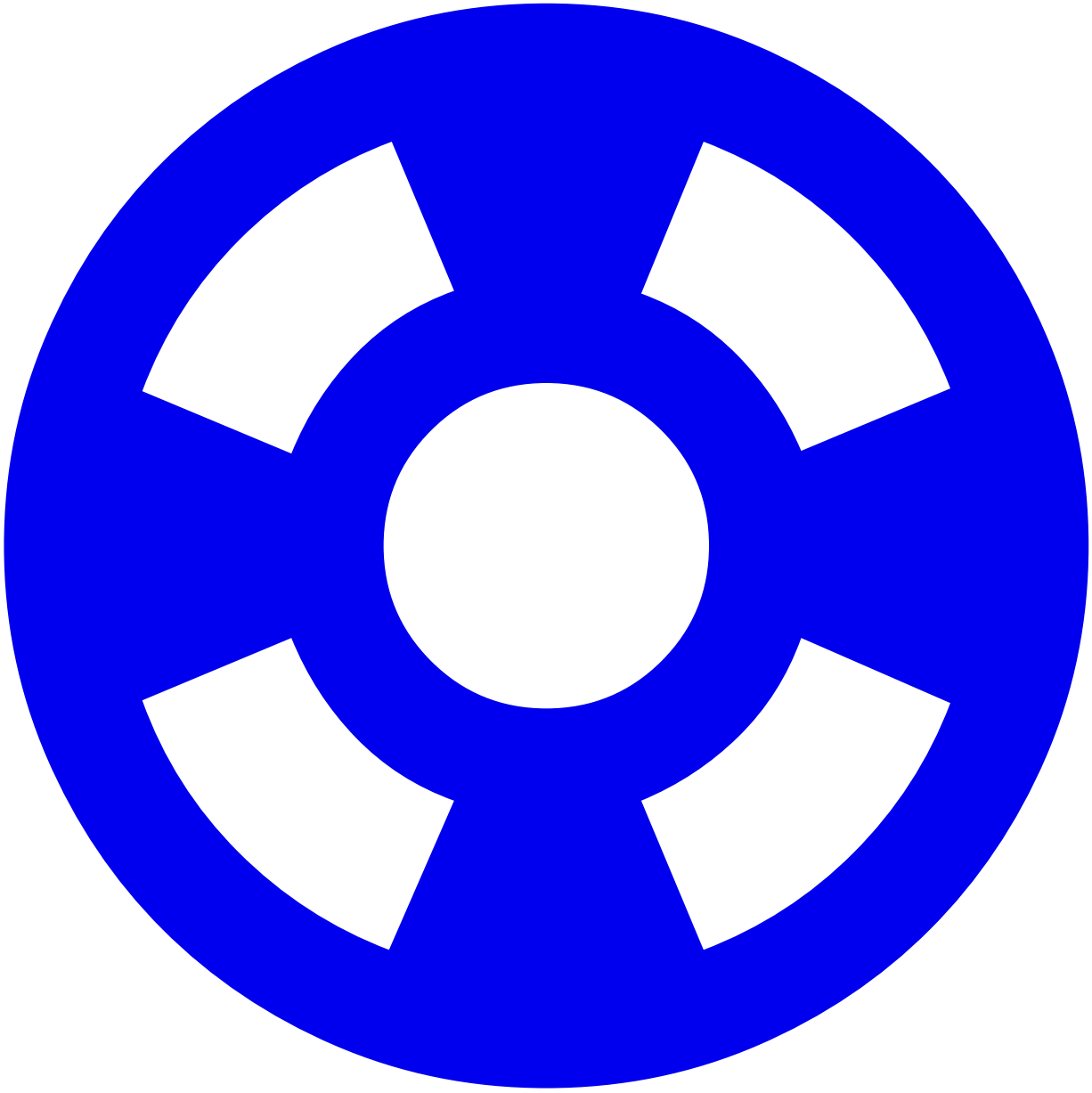


We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



C

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

T

K ly connecting, governing, and monetizing APIs and AI tokens across any model or cloud.

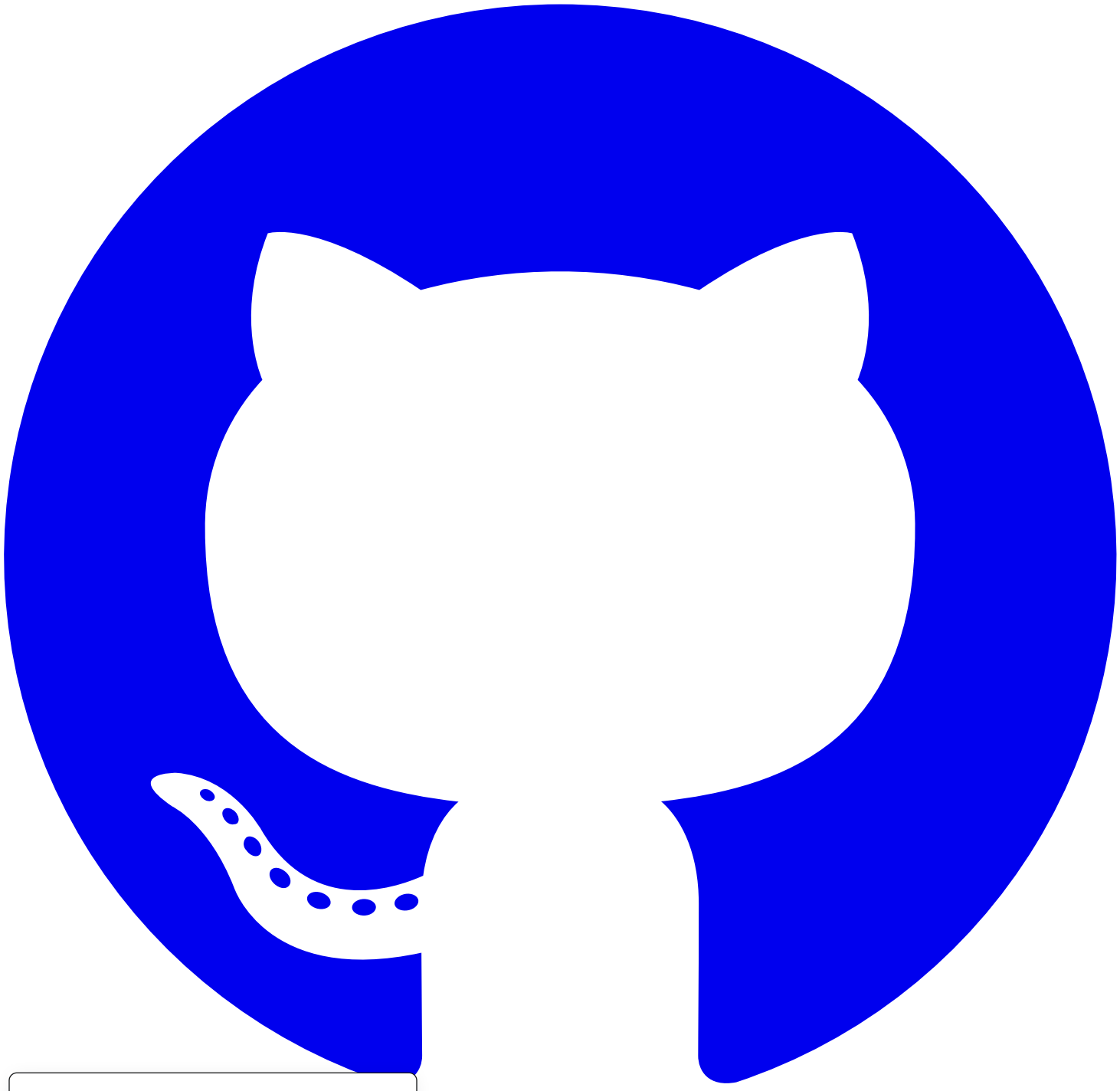
R
S
P
A al Pricing
P
K PI Specs Ingress Controller
P
A lugins Traffic Control Plugins View all →
H
K
S

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

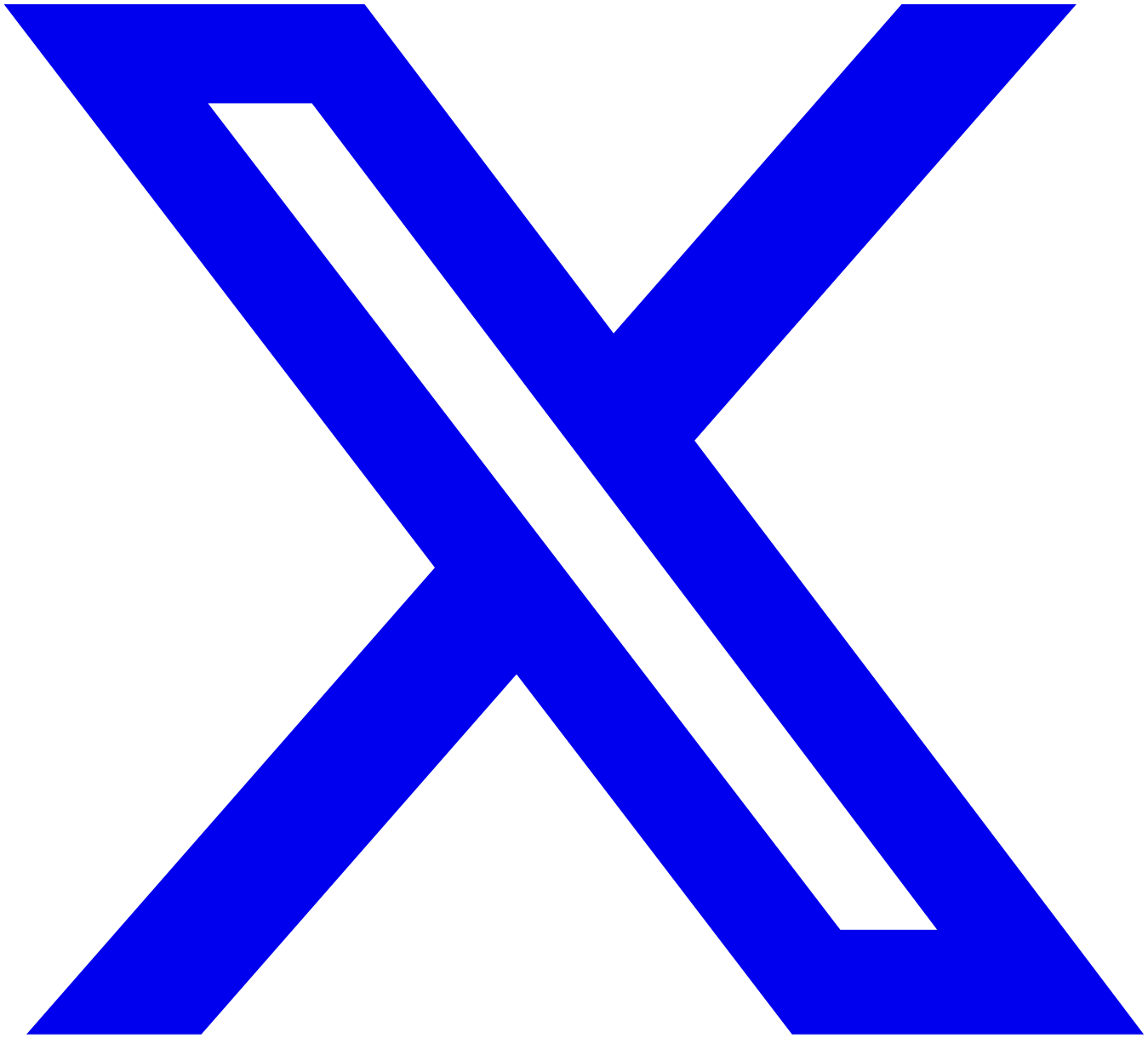


We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)

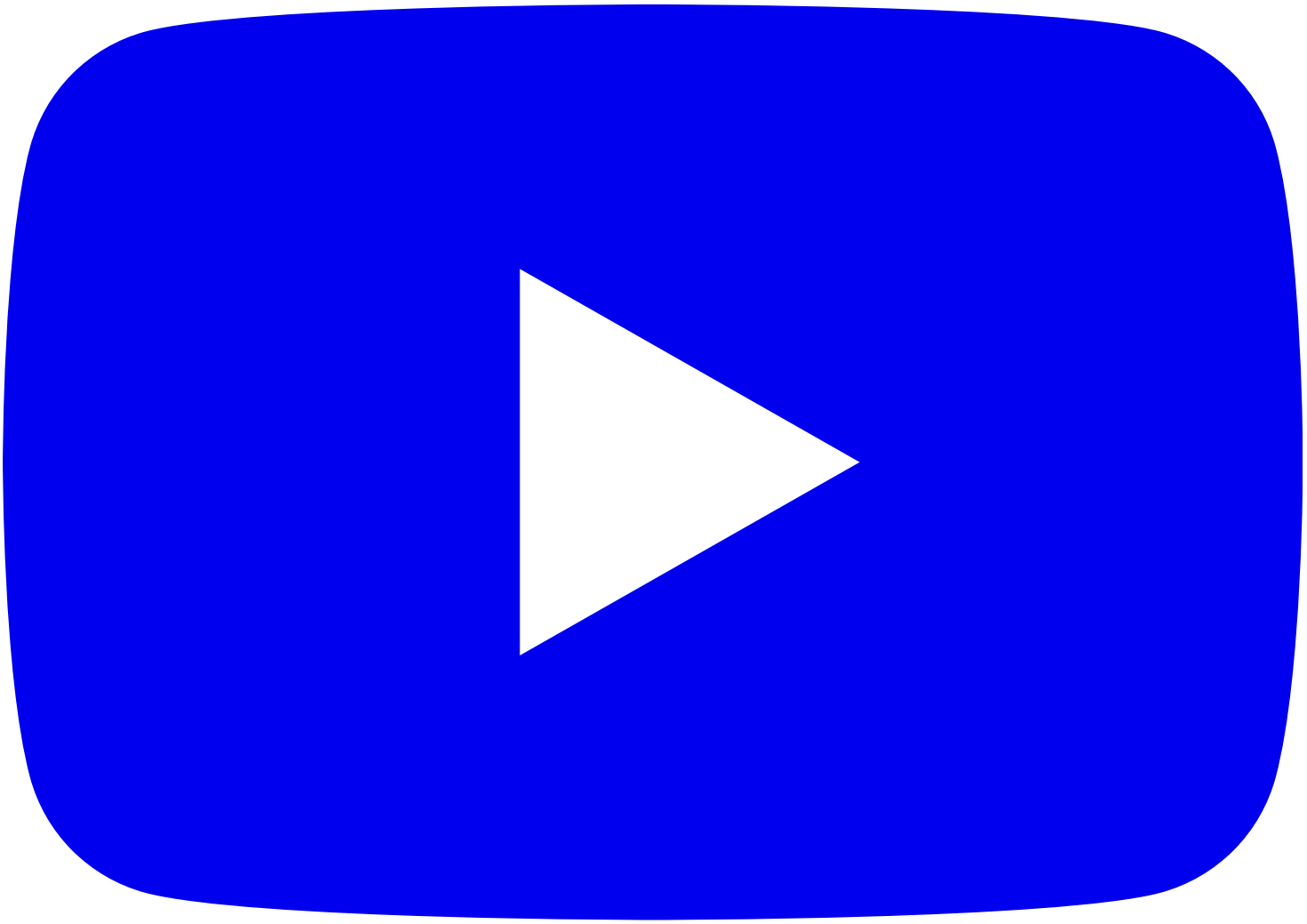


We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)



T
©

We Value Your Privacy

English

By clicking "Accept All", you agree to the storing of cookies to enhance site navigation, usage and marketing efforts.

(Privacy Policy)