

Conduktor equips Platform teams to

- ✓ Enforce security standards & rules
- ✓ Provide visibility into Kafka
- ✓ Delegate ownership & autonomy to developers



Encrypt your data with ease

Simplify the encryption landscape. Introduce a unified security layer across your Kafka, removing the need for teams to implement a solution on every application

Scalable Governance

Accelerate cross-team data sharing. Build applications quicker in a data-mesh architecture by having platform teams empower developers to self-serve requests for the data & Kafka resources that they own

Protect your cluster

Prevent mistakes by enforcing configuration. Topic creation, producing messages and nearly every other area of Kafka can be safeguarded by your choice of requirements

Democratize your Kafka data, securely

Lower the entry barrier on Kafka. Anyone can securely explore topics and find their messages using the best UI, with appropriate access

One place to manage all your Kafka resources

Control your entire Kafka ecosystem in one place. Manage records, topics, schema registries, consumer groups, connectors, service accounts and more

Proactively Monitor, Alert and Audit

Take preemptive measures to monitor and notify on your cluster and streaming applications. Audit everything for traceability of user and application actions to meet compliance standards.

USED BY:

Architects

Software Developers

Platform Administrators

DevOps



Vendor agnostic
We support Confluent, MSK, Redpanda, Aiven & more!



Security and Encryption



Visibility into Kafka



Self-service with Governance



Console

Back

ADVANCED

Clusters

Certificates

Data policies

YOUR ORGANISATION

Users

Groups

Audit log

Integrations

API Keys

Plan

ACCOUNT

Profile



Console

Cluster pkc-yytws
solitary-paper

MAIN

Home

Topics

Schema registry

Consumer groups

Kafka connect

Brokers

Service accounts

Gateway interceptors

ksqlDB

MONITORING

Overview



Search resources, users, groups and actions



Search resources, users, groups and ac

Topics

Search by name

Internal top

Consumer group ↑

sales-performance-analytics-report

team-blue team-violet

customer-feedback-survey-results

data-engineers

inventory-management-system-status

employee-time-tracking-records

data-engineers production

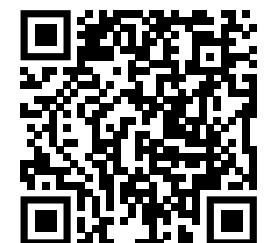
machine-learning-model-training

employee-time-tracking-records

analytics

Try Conduktor for free, or book a demo!

conduktor.io/get-started





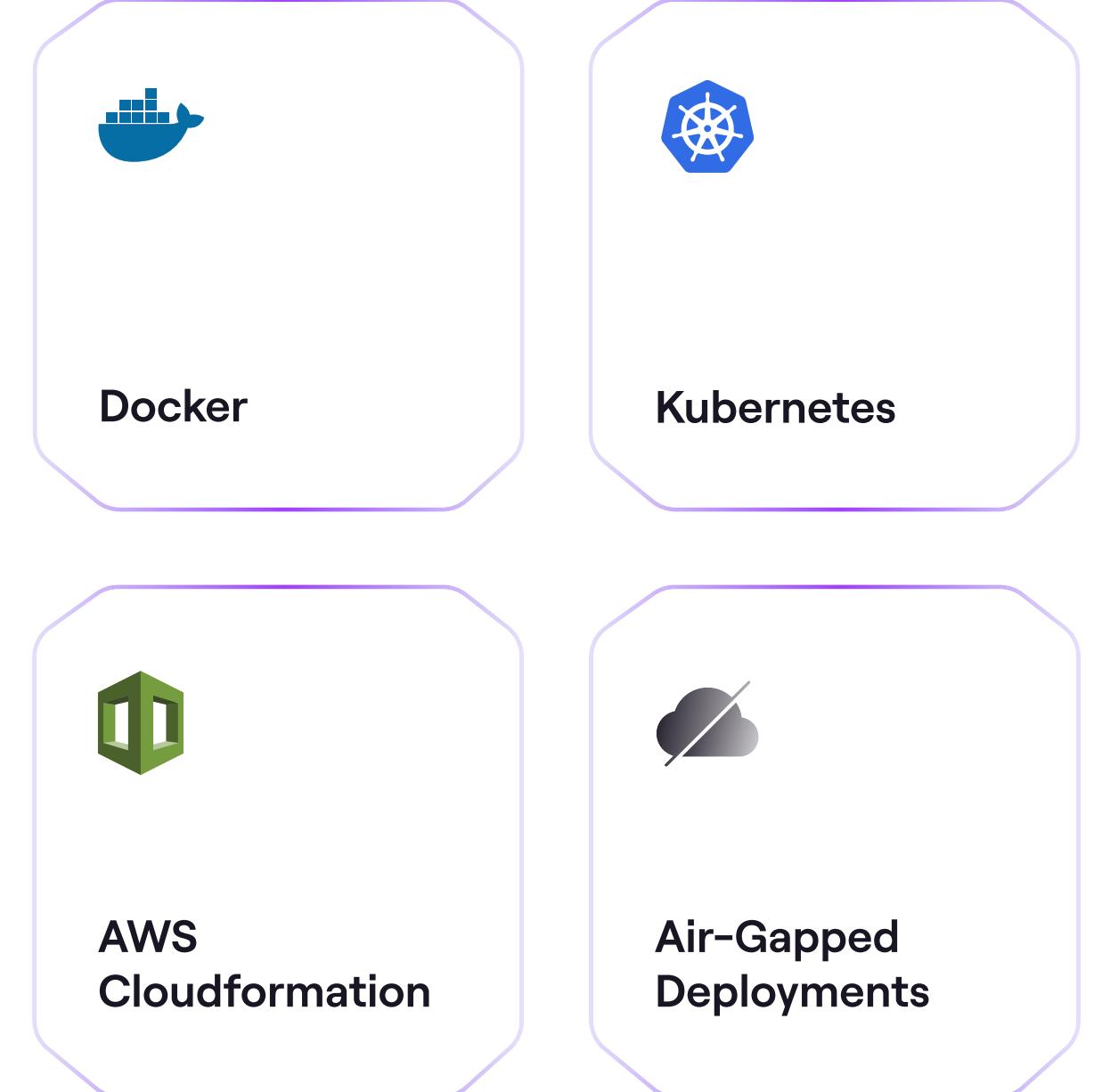
Conduktor Gateway is a **Kafka proxy** deployed between your Kafka and your applications. As it speaks the **Kafka protocol** there is minimal adjustments beyond a bootstrap server change.

Deployments

Scalable: A Docker image that can be deployed like any other within your containerisation setup, such as on Kubernetes or managed services.

Resilient: Like Kafka if a broker dies it can be restarted whilst Gateway keeps running. As Kafka protocol magic is at play your applications remain available.

Latency: By default GW is unnoticeable, it will add latency in the order of milliseconds. If you start to add heavier functionality, like encryption, there is naturally a small price to pay.



Console

Find messages easily. Search, sort and filter messages easily in our UI by the key, value, headers, and all other associated metadata.

Your permissions over Kafka. Manage access for users & groups to all Kafka resources with our granular RBAC model. Bring existing groups from your identity provider!

Support all your Kafka ecosystem. Conduktor is multi-cluster, vendor agnostic. Connect to Cloud or on-prem clusters as well as ksqlDB, Connect, Schema Registry and more.



Data Security

Message and field-level encryption. Encrypt messages centrally for all projects with a flexible, language & vendor agnostic solution.

Audit everything. Audit some or all of your applications interacting with your Kafka clusters.

Encrypt when sharing with partners. Meet standards with confidence by ensuring all messages that are shared must be encrypted.



Governance

Self-serve project teams. Remove your Platform Ops bottleneck, give appropriate responsibility for access requests & resource creation to the team that owns the data.

Enforce Kafka best practice. Prevent unintended consequences by enforcing configuration such as use of compression on produce, naming convention & partition count when creating topics, or any other area of Kafka you need to govern.

Prevent data quality issues at produce. Prevent issues before they arise by validating the quality of messages before they hit the Kafka cluster, and the rest of your infrastructure .



Optimize

Maximise utilization. Curb the rise of partition count with unlimited partitions, leveraging one physical topic to power many different applications.

Avoid unnecessary data replication. Leverage Conduktor's multi-tenancy architecture to create true isolation without needing to replicate to additional infrastructure.

Handle large messages in Kafka. Avoid load spikes and network overhead by offloading large messages to object storage to reclaim as needed.

