CASE STUDY

SQUARESHIFT

A global video commerce company migrates Elasticsearch to Elastic Cloud with zero downtime

SERVICE Premier Partner Google Cloud

SPECIALIZATION Machine Learning Google Cloud

SPECIALIZATION Data Analytics Google Cloud

30%
Estimated savings in storage costs
→ By skipping unnecessary data transfer and optimizing retention

95%
Time saved on alert updates

3500 +
Watchers updated via automation
instead of manual edits

Client

The client is a global leader in video commerce, operating across TV, e-commerce, and mobile platforms. Their Elasticsearch environment powers critical observability and alerting workflows.

Facing high ingestion volumes and the need for secure, scalable infrastructure, they migrated system indices to Elastic Cloud while implementing dual writes with Logstash and Fleet for real-time continuity.

At-A-GLANCE

Challenges

- Snapshot restoration failed due to credential issues
- ILM misconfigured for unexpected data ingestion
- API decryption errors disrupted alerting
- SAML access blocked dashboards and reports
- Log ingestion failed from template mismatches

Solutions

- **Snapshot Access Fixed**: Resolved storage credential issues through manual verification

- **ILM Tuned**: Adjusted tiering and policies to handle ingestion spikes and retention needs

- **API/SAML Resolved**: Fixed decryption bugs and adjusted roles for full dashboard/report access

- **Template Corrections**: Aligned index templates to restore accurate log ingestion and monitoring visibility

Project Context

The client had an on-prem Elasticsearch setup powering log ingestion and alerting through Logstash, Fleet, and Kibana. With a short data retention policy and fast ingestion rates, they opted for a selective migration, transferring only system indices and enabling dual writes to Elastic Cloud. This minimized risk, reduced cost, and avoided transferring 52TB of historical data.

Project Objectives

- Migrate system indices (users, templates, policies) to Elastic Cloud
- Maintain live operations using dual writes for Fleet and Logstash
- Configure secure authentication via SAML and API keys
- Enable observability through real-time monitoring setup
- Optimize ILM and storage to align with high ingestion rates

Solution Delivery

Phase 1: Preparation & Snapshot Setup

- Reviewed Logstash configs and dual write setup
- Took snapshots of critical system indices (.watcher, .security, .kibana)
- Created non-prod Elastic Cloud deployment and validated storage access.

Phase 2: Dual Writes and Production Cutover

- Enabled Logstash dual writes without queuing risk
- Migrated 4 Fleet agent policies and tuned ILM for hot/frozen tiers
- Deployed monitoring cluster, integrated Logstash and Metricbeat
- Automated update of 3515 Watcher URLs via Python script

Phase 3: Optimization and Issue Resolution

- Resolved API decryption, Fleet auth, and ILM misconfigurations
- Fine-tuned audit log indexing and dashboard access controls
- Addressed NAT IP and SLM repository issues for connectivity

Technology Stack

elasticsearch | elastic cloud | Microsoft Azure Blob Storage | logstash | kibana | SAML 2.0