

Project Horizon

Customer Data Platform (CDP) Implementation Report

1. Executive Summary

This Proof of Concept (PoC) demonstrates the foundational capabilities of a **Customer Data Platform (CDP)** designed to capture, analyze, and act upon user activity data. The objective was to simulate customer behavior, assess churn risk, and trigger targeted engagement campaigns with full observability into the process.

The solution leverages a lightweight, open-source stack consisting of **Postgres**, **Python services**, **Mailgun SMTP**, and the **ELK Stack**. Together, these components provide an end-to-end workflow: user activity simulation, churn scoring, alert creation, campaign email delivery, and centralized log monitoring.

Key achievements include:

- **Event Simulation and Storage:** Synthetic user events generated and persisted in Postgres.
- **Churn Scoring:** Rule-based scoring model applied on recent user activity to identify at-risk users.
- **Alerting and Campaigns:** High-risk users automatically flagged with alerts and targeted campaigns (“miss you” and “good activity”) sent via Mailgun.
- **Observability:** JSON-based logs ingested into ELK, enabling real-time dashboards for monitoring pipeline stages, campaigns, and outcomes.

This PoC validates the feasibility of building a CDP using open-source and easily extensible components. While core functions simulation, scoring, alerting, emailing, and monitoring have been delivered successfully, certain enterprise features such as **identity resolution, preference management, advanced segmentation, and machine learning-based personalization** remain outside current scope. These are proposed as part of the roadmap to scale the platform into a production-ready CDP.

2. Introduction

Modern organizations require a **Customer Data Platform (CDP)** to unify customer activity, analyze engagement patterns, and activate personalized campaigns in real time. The requirements were framed around building a prototype solution capable of:

- **Ingesting customer activity data** in a structured and observable way.
- **Deriving engagement insights and churn risk scores** based on activity levels.
- **Triggering automated alerts and campaign actions** to re-engage at-risk users.
- **Providing full observability and auditability** through centralized logging and dashboards.
- **Maintaining alignment with compliance requirements** related to customer engagement and communication.

To meet these requirements within the scope of a PoC, a **lightweight, open-source stack** was adopted, focusing on modularity and extensibility. Instead of integrating with live customer-facing applications, the solution generates **synthetic activity data** to simulate user behavior patterns. This approach allows validation of scoring, alerting, and campaign automation without dependency on production systems.

The delivered prototype demonstrates the following:

- A **simulation pipeline** that generates synthetic logins, activity, and inactivity events.
- A **scoring mechanism** that classifies users based on their recent activity and inactivity trends.
- An **alerting layer** that flags users crossing configurable risk thresholds.
- A **campaign activation mechanism** that delivers targeted emails through Mailgun using professional HTML templates.
- An **observability layer** that ingests all pipeline logs into ELK, enabling dashboards for tracking alerts, campaigns, and outcomes.

This introduction establishes the baseline: the CDP PoC is not a full-scale production deployment, but a **validated prototype** demonstrating how synthetic activity can be transformed into actionable insights and automated engagement campaigns.

3. Scope

3.1 Current Scope (PoC Coverage)

The current CDP Proof of Concept (PoC) delivers a streamlined pipeline that demonstrates how synthetic customer activity can be **ingested, scored, alerted, and activated into campaigns**. Specifically, it includes:

- **Data Simulation:** Synthetic user activity generation (logins, inactivity, churn-risk scenarios).
- **Scoring Engine:** Engagement scoring with configurable thresholds (high, medium, low) based on activity windows.
- **Alerts:** Automated alert generation for at-risk users, logged into Postgres and made available for campaign triggers.
- **Campaign Activation:** Integration with Mailgun for email campaigns using two professional templates:
 - “Miss You” template for low-activity users.
 - “Good Activity” template for healthy engagement users.
- **Observability & Monitoring:** Unified log collection into **Elasticsearch (ELK)** with dashboards to visualize pipeline stages, alerts, and email campaigns.
- **Demo Campaign Mode:** Ability to scope alerts to a single recipient for demonstration purposes, ensuring realistic but safe campaign testing.

3.2 Constraints

The PoC is designed for demonstration and validation purposes. Key constraints include:

- **Synthetic Data:** No real customer data was used; all activity and engagement scores are simulated.
- **Email Campaign Limitation:** Campaigns are limited to Mailgun integration and a controlled set of templates.
- **Minimal Infrastructure:** Deployment is limited to VM-based setup on local/PoC infrastructure (Postgres, ELK, Mailgun), without enterprise-scale cluster support.
- **Campaign Focus:** Prototype demonstrates re-engagement campaigns, but not the full spectrum of CDP campaign types (cross-sell, upsell, segmentation-driven campaigns).

3.3 Gap Analysis

When compared to a production-grade CDP, the following gaps exist in the current PoC:

- **Lineage & Full User Journey Mapping:** Current solution scores activity but does not track multi-touch user journeys.
- **Segmentation & Personalization:** Templates are static, lacking advanced personalization based on user attributes.
- **Advanced Policy Controls:** No built-in compliance enforcement for opt-in/opt-out tracking beyond PoC email demo.
- **Scalability:** Designed for single-node PoC, not multi-region or high-volume data pipelines.

3.4 Future Enhancements

To achieve a fully production-ready CDP, the following features are recommended:

- **User Segmentation Engine:** Grouping users by demographics, behaviors, and engagement attributes.
- **Personalized Campaign Orchestration:** Dynamic email templates with variable substitution and A/B testing.
- **Lineage & Analytics:** Advanced lineage tracking with Apache Atlas or similar tooling for auditability.
- **Enterprise-Grade Compliance:** Consent management, opt-out policies, and GDPR/CCPA compliance integration.
- **Scalable Infrastructure:** Migration to Kubernetes and cloud-native storage for handling high-volume events.
- **Multi-Channel Support:** Expanding campaigns beyond email to include SMS, push notifications, and in-app alerts.

4. Tools & Solution Overview

The CDP PoC leverages a **modular open-source stack** integrated into a single engagement pipeline. Each component plays a distinct role while working together to deliver end-to-end functionality.

4.1 PostgreSQL (User & Event Store)

- Acts as the **central metadata store** for user activity, engagement scores, alerts, and campaign notifications.
- Stores synthetic user profiles (user_id, activity logs) and campaign-related events.
- Enables historical queries, reporting, and auditing of engagement activity.

4.2 Data Generator & Scorer

- **Generator Module** simulates user activity (logins, inactivity windows, churn risks).
- **Scorer Module** calculates user engagement scores within a rolling window (14 days), classifying into **high**, **medium**, or **low** tiers.
- Provides the **input feed** for alerting and campaign decisions.

4.3 Alerter

- Monitors engagement scores and applies **threshold-based rules** to trigger alerts (e.g., score < 0.5 = medium, score < 0.7 = high-risk).
- Inserts alerts into the database for further processing.
- Ensures **cooldown windows** (7 days) are enforced to avoid spamming users.

4.4 Mailgun (Email Delivery)

- Provides the **transactional email service** used in the PoC.
- Integrated with campaigner modules to send:
 - “**Miss You**” Template (targeting low-activity users).
 - “**Good Activity**” Template (rewarding engaged users).
- Supports both **bulk campaigns** and **demo mode** (single recipient for safe testing).

4.5 Demo Campaigner

- A special-purpose utility enabling **demo campaigns** without real user data.
- Replaces synthetic user email addresses with a **live test email** provided by the operator.

- Ensures alerts can be safely tested, one at a time, for **demonstration and validation** purposes.

4.6 Log Pipeline & Elasticsearch (ELK)

- All CDP components emit structured logs into a unified `cdp_events.log`.
- A custom **log shipper script** forwards these logs into **Elasticsearch**.
- **Kibana Dashboards** visualize pipeline health and campaign activity:
 - Ingested events per run.
 - Engagement score distribution.
 - Alerts generated vs. skipped.
 - Campaign emails sent vs. failed.

5. Configurations & Customizations

The CDP PoC required several custom configurations to adapt open-source tools and make them interoperate as a single pipeline.

5.1 PostgreSQL

- **Custom Tables:**
 - `current_scores` → stores latest engagement scores for each user.
 - `alerts` → logs triggered alerts with status (NEW, NOTIFIED).
 - `notifications_log` → records campaign emails sent (user, campaign ID, status).
- **Schema Adjustments:**
 - `alert_id` column modified to allow NULL for demo campaigns (flexibility in testing).

5.2 Data Generator & Scorer

- Configurable **activity windows** (`WINDOW_DAYS=14`) and **thresholds**:
 - High risk: $\text{score} \geq 0.7$
 - Medium: $0.5 \leq \text{score} < 0.7$

- Low: < 0.5
- **Synthetic data** created for ~200 users with randomized login frequencies and churn behavior.

5.3 Alerter

- **Cooldown Policy:** Configured for 7 days to avoid repetitive alerts on the same user.
- **Threshold Flexibility:** Can be re-run with adjusted score thresholds to simulate different campaign strategies.

5.4 Mailgun Integration

- **Environment Variables:** Configured for SMTP host, port, user, and password.
- **Templates:**
 - miss_you.html for churn-risk users.
 - good_activity.html for engaged users.
- **Demo Mode:** Extended with demo_campaigner to override synthetic email addresses with a real recipient for controlled testing.

5.5 Log Pipeline → Elasticsearch

- **Log File Path:** Centralized at logs/cdp_events.log.
- **Log Fields:** Normalized to a fixed schema (@timestamp, stage, component, campaign_id, status, etc.).
- **Custom Shipper Script:**
 - Performs **initial ingestion** of historical logs.
 - Uses **watchdog monitoring** to stream new log lines in real time.
 - Sends structured JSON to Elasticsearch under the index cdp_events.
- **Kibana Dashboards:** Configured to visualize:
 - Engagement scores.
 - Alerts created vs. skipped.

- Campaigns sent vs. failed.

6. Architecture & Network Diagram

The CDP PoC is designed as a modular but connected pipeline where **synthetic user activity** is generated, scored, alerted upon, and finally results in **campaign emails**. All logs are centralized into ELK for monitoring.

6.1 Architecture Flow

1. Synthetic Data Simulation

- Generates user activity patterns (logins, inactivity).
- Feeds events into PostgreSQL.

2. Scorer

- Computes rolling engagement scores for each user.
- Stores results in `current_scores`.

3. Alerter

- Compares scores against thresholds.
- Creates alerts in alerts table when risk levels are met.
- Enforces cooldowns to prevent duplicate alerts.

4. Campaigner (Mailer)

- Fetches alert-triggered users.
- Sends appropriate campaign email using Mailgun:
 - Miss You Template → for inactive users.
 - Good Activity Template → for active users.
- Logs results in `notifications_log`.

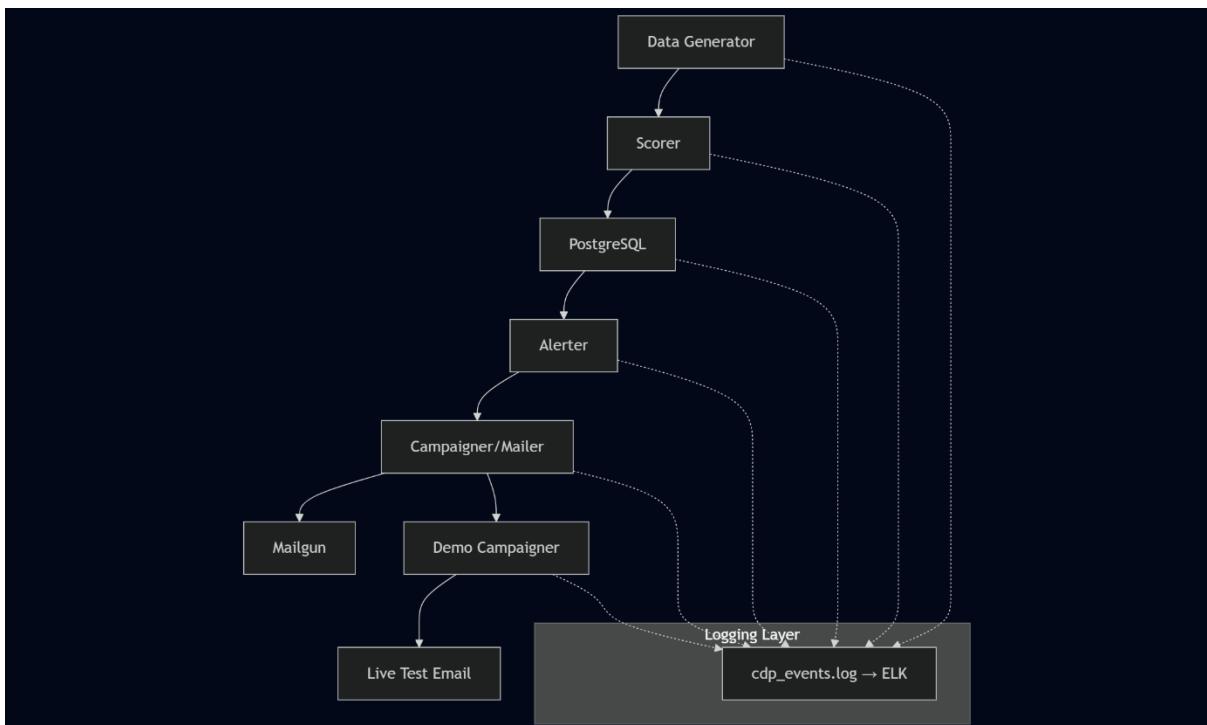
5. Demo Campaigner (Testing Mode)

- Overrides synthetic emails with a **live test email address**.
- Sends one alert at a time for safe demo purposes.

6. Log Aggregation (Pipeline → ELK)

- All components write structured events to `cdp_events.log`.
- A Python log shipper parses and forwards logs to **Elasticsearch**.
- Kibana dashboards present visual insights into system health and campaign activity.

6.2 Network Diagram



7. Evidence & Proof of Concept

The following artifacts demonstrate that the CDP PoC is fully functional, covering the lifecycle from **data generation** → **scoring** → **alerting** → **campaign delivery** → **log centralization**.

7.1 Database Evidence

- **Schema & Seed Data**
 - Successfully created schema tables: `users`, `events`, `current_scores`, `alerts`, `notifications_log`.
 - Seeded with 200 synthetic users and randomized activity patterns.

- **Query Output**

```
root@CDP:/home/platform# psql "$PG_DSN" -c "SELECT user_id, score FROM current_scores ORDER BY score DESC LIMIT 5;"
```

user_id	score
user_088	0.9285760084069113
user_058	0.500045798354829
user_190	0.500045798354829
user_155	0.35753050322503305
user_007	0.35714743697834

(5 rows)

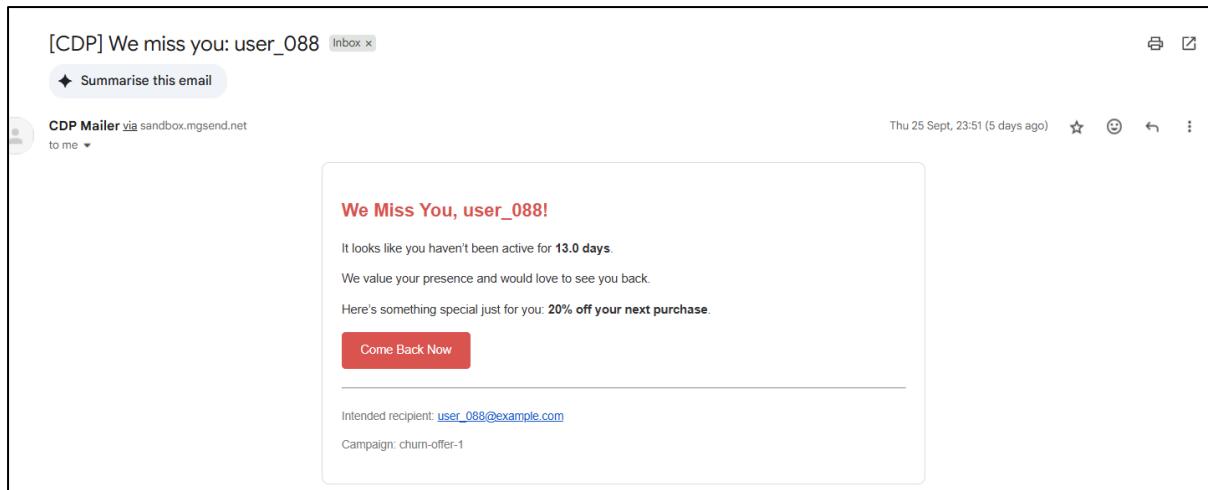
7.2 Scoring & Alert Logs

- **Scorer Component** produced tiered results.
- **Alerter Component** generated alerts when thresholds exceeded.

```
root@CDP:/home/platform/cdp-prototype-phase1/cdp-prototype# tail -n 50 logs/cdp_events.log
{"ts": "2025-09-25T12:25:27Z", "level": "INFO", "message": "ingest", "stage": "ingest", "component": "simulator", "correlation_id": "8dd6431d-7aa9-4dc2-bbeb-ed686b81250b", "inserted": 633, "users": 188}
{"ts": "2025-09-25T12:25:27Z", "level": "INFO", "message": "score", "stage": "score", "component": "scorer", "correlation_id": "1778e24e-a0e7-48b4-8d33-937a72ab6e7d", "window_days": 14, "users": 202, "high": 14, "medium": 0, "low": 188}
{"ts": "2025-09-25T12:25:27Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "643eaf56-7c49-48b9-b7fa-ce8fd43d1925", "created": 14, "threshold": 0.7, "cooldown_days": 7}
{"ts": "2025-09-25T12:25:28Z", "level": "INFO", "message": "email", "stage": "email", "component": "mailer", "correlation_id": "c12b458d-9cd5-4016-a9b2-e0666ff9fcc5", "campaign_id": "churn-offer-1", "sent": 0, "skipped": 14, "failed": 0}
{"ts": "2025-09-25T12:26:59Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "e07327ea-1bd3-4385-bbb7-84b60f2144cc", "created": 14, "threshold": 0.5, "cooldown_days": 7}
{"ts": "2025-09-25T12:26:59Z", "level": "INFO", "message": "email", "stage": "email", "component": "mailer", "correlation_id": "3e36d494-75d7-4f62-b6ee-9af4a9a4316a", "campaign_id": "churn-offer-1", "sent": 0, "skipped": 0}
{"ts": "2025-09-25T12:28:28Z", "level": "INFO", "message": "score", "stage": "score", "component": "scorer", "correlation_id": "b70ba785-2b5d-41a5-9282-7a7380491062", "window_days": 14, "users": 202, "high": 14, "medium": 0, "low": 188}
{"ts": "2025-09-25T12:28:28Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "5fe8ecc4-7467-4450-b263-150b725de253", "created": 0, "threshold": 0.7, "cooldown_days": 7}
{"ts": "2025-09-25T12:33:11Z", "level": "INFO", "message": "ingest", "stage": "ingest", "component": "simulator", "correlation_id": "99445430-7ee2-4928-bacf-535faf704e4b", "inserted": 2783, "users": 202}
{"ts": "2025-09-25T12:33:16Z", "level": "INFO", "message": "score", "stage": "score", "component": "scorer", "correlation_id": "ea98898e-d32f-4729-974a-fac3ce45bde1", "window_days": 14, "users": 202, "high": 1, "medium": 2, "low": 199}
{"ts": "2025-09-25T12:33:22Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "a03f2c17-af39-45fd-835a-adc8dd69fee", "created": 0, "threshold": 0.7, "cooldown_days": 7}
{"ts": "2025-09-25T12:33:26Z", "level": "INFO", "message": "email", "stage": "email", "component": "mailer", "correlation_id": "716311aa-34ce-4ea6-9913-5becf291817d", "campaign_id": "churn-offer-1", "sent": 0, "skipped": 0, "failed": 0}
{"ts": "2025-09-25T12:50:50Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "9322389a-a66f-4b25-d02-1819fdb4e967", "created": 0, "threshold": 0.7, "cooldown_days": 7}
{"ts": "2025-09-25T12:52:16Z", "level": "INFO", "message": "alert", "stage": "alert", "component": "alerter", "correlation_id": "4b67d75e-b68e-414a-b4d6-93e95362cdb2", "created": 0, "threshold": 0.7, "cooldown_days": 7}
{"ts": "2025-09-25T12:52:18Z", "level": "INFO", "message": "email", "stage": "email", "component": "mailer", "correlation_id": "bad1c029-5407-463e-902e-3ef805cea8a2", "campaign_id": "churn-offer-1", "aggregated": true, "to_addr": "jainyogendral685@gmail.com", "count": 0, "status": "skipped", "error": null}
{"ts": "2025-09-25T12:53:43Z", "level": "INFO", "message": "email", "stage": "email", "component": "mailer", "correlation_id": "480c1fa9-56db-4421-911b-6b9713dd43c9", "campaign_id": "churn-offer-1", "aggregated": true, "to_addr": "jainyogendral685@gmail.com", "count": 28, "status": "sent", "error": null}
```

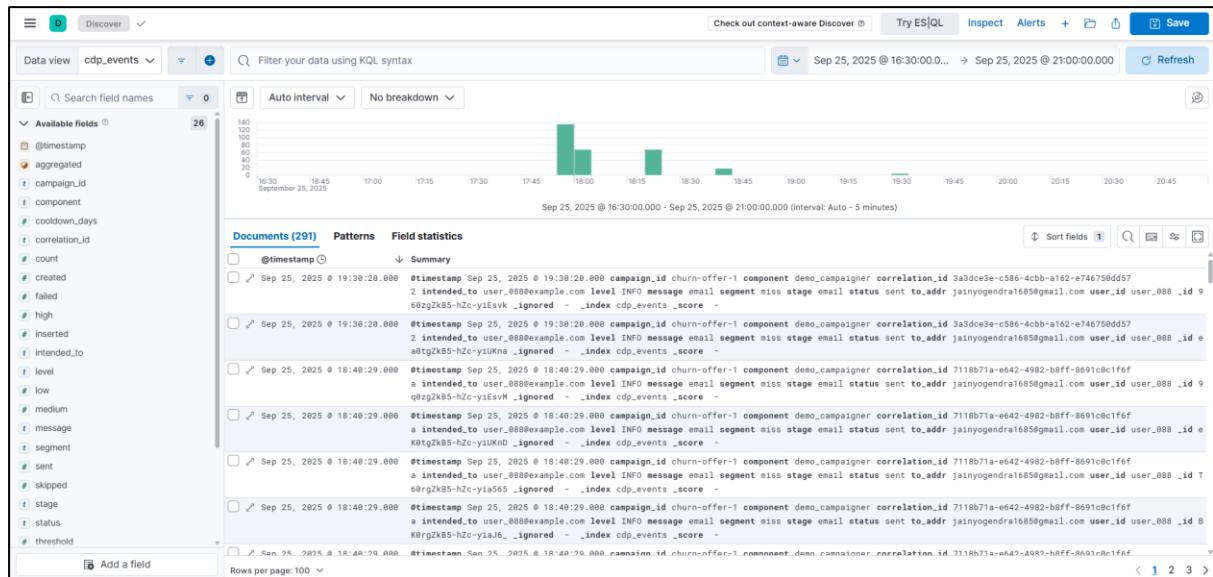
7.3 Campaign Evidence

- **Aggregated Campaign Run**
- **Demo Campaigner Run (Live Mail)**
- **Received Email Proof**
 - “Miss You” template successfully received in live Gmail inbox.
 - Confirms end-to-end mail delivery via Mailgun integration.

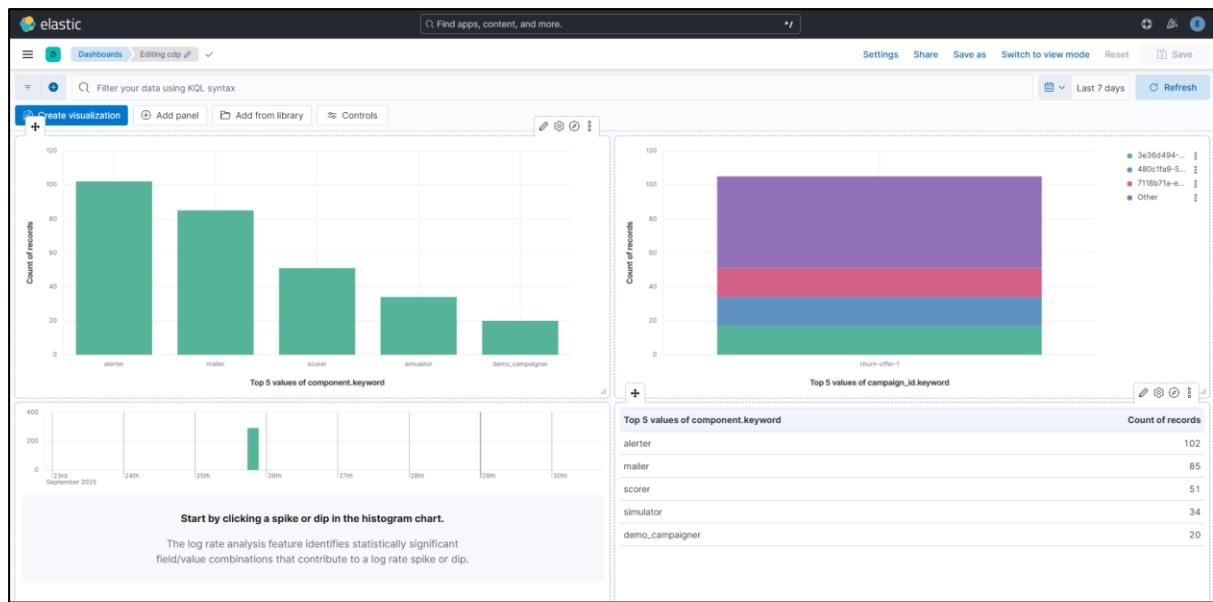


7.4 ELK Dashboard Evidence

- Ingestion Script** shipped cdp_events.log into Elasticsearch index cdp_events.



- Verified records show fields like user_id, segment, status, campaign_id.
- Dashboards highlight:
 - Email status distribution (sent, skipped, failed).
 - Top users by score.
 - Timeline of campaign activity.
- Attached screenshot: ELK dashboard visualizing email campaign logs and activity distribution.



7.5 End-to-End Proof

The PoC demonstrates the full flow:

- Synthetic data simulated → Alerts generated.
- Campaigns executed (both synthetic and live test).
- Logs ingested into ELK → dashboards available for SOC-style monitoring.

8. Compliance Mapping

Although CDP is a Proof of Concept and not a production-grade compliance product, its components map to several established frameworks and standards. The following table highlights how current capabilities align with requirements:

Control Area	Standard / Reference	CDP Capability	Notes / Gaps
User Activity Monitoring	ISO 27001 A.12.4 (Logging & Monitoring), NIST CSF DE.CM	Ingestion + scoring engine tracks inactivity/engagement windows. Alerts created when thresholds exceeded.	Currently limited to synthetic events; production would require integration with real user activity logs.
Notification & Response	ISO 27001 A.16 (Incident	Alerter + Mailer components automatically	Presently limited to churn/engagement

	Management), GDPR Art. 34 (Communication of Breach)	notify (via campaign) when conditions are met.	scenarios, not security breaches. Templates can be adapted for compliance alerts.
Data Integrity & Auditability	ISO 27001 A.12.7 (Event Logging), NIST SP 800-53 AU-3	PostgreSQL notifications_log and Elasticsearch cdp_events indexes capture immutable records of all alerts and notifications.	No log retention/encryption policies defined yet.
Awareness & Preventive Action	NIST CSF PR.AT, GDPR Art. 25 (Privacy by Design)	Campaign templates ("miss you", "good activity") demonstrate preventive communication.	Needs extension to security awareness or compliance reminders in a real environment.
Monitoring & Detection	MITRE ATT&CK Detection Engineering	ELK dashboards provide monitoring of campaign activity, user scores, and system health.	Coverage limited to engagement scoring; not mapped to broader threat detection yet.

Key Observations

- **Strengths:** CDP already covers logging, scoring, alerting, and notification workflows showing clear alignment with monitoring and response domains.
- **Gaps:** Lacks enterprise-grade lineage, encryption at rest/flight, and integration with IAM/KMS needed for compliance-heavy environments.
- **Scalability:** Current architecture can be extended to compliance-sensitive use cases (e.g., data retention alerts, insider threat notifications) with minimal changes to pipelines.

9. Conclusion

The **Customer Data Platform (CDP) Proof of Concept** successfully demonstrates how an integrated pipeline of open-source tools can be leveraged to ingest, process, score, and act on user activity data. The project delivered:

- A **PostgreSQL-backed ingestion and scoring model** that computes engagement scores and identifies users at risk of churn or inactivity.
- An **alerting engine** that transforms scoring results into actionable notifications.
- A **mailer and demo campaigner workflow** capable of sending contextual, template-based communications to designated recipients.
- **Centralized observability through ELK**, ensuring that all CDP events ingest, score, alert, notify-are logged and monitored.

Achievements

- End-to-end pipeline tested with synthetic user data.
- Real email notifications validated via Mailgun integration.
- ELK dashboards provide full visibility into engagement events and alerts.
- Extensible modular design allowing future campaigns.

Limitations & Gaps

- Current setup runs only on **synthetic data**; production would require real integration with applications.
- **Security hardening** (TLS everywhere, key management, RBAC) is not yet implemented.
- Compliance coverage is partial requires lineage, retention policies, and integration with enterprise IAM/KMS for production.

Roadmap for Production

1. Integrate with **real user activity sources** (CRM, product logs, identity systems).
2. Extend campaign templates to cover **compliance and risk communications**.
3. Add **data lineage and retention controls** for regulatory alignment.
4. Introduce **enterprise IAM and key management** for secure operations.
5. Harden infrastructure with **Zero Trust access, TLS certs, and monitoring**.