**🚗 Driver for the PII Logging Library**

The **primary driver** for developing this library is the **TechRisk compliance requirement** that mandates **logging of all user access to sensitive personal data (PC2/PC3 classified fields)** in any application that processes PII.

As per TechRisk guidelines:

* Applications **must generate audit events** whenever PII data is accessed API.
* Logs must adhere to a **strict JSON format** and should **exclude actual sensitive values**.
* Logs must be **retained for at least 3 months** and must include metadata such as event time, user ID, source IP, and type of PII accessed.

**🔍 Existing Solutions and Why They Are Not Suitable**

**✅ Available Central Solutions for PII Logging**

1. Atlas PII Logging Library
2. Central Audit Logging Library

**🚫 Limitations of Central Solutions**

| **Limitation** | **Description** |
| --- | --- |
| **❌ No Protobuf Support** | **These libraries do not support Protobuf v3, which is the primary data modeling format in our applications.** |
| **❌ Annotation Mismatch** | **The @PiiDataAccess annotation model is not compatible with Protobuf-generated classes.** |
| **❌ Framework Limitation** | **Both libraries were built with Jersey + POJO assumptions and require substantial adaptation to work with Spring Boot or gRPC.** |
| **❌ No BQL Support** | **The existing central libraries do not support writing logs to BQL, which is our team's preferred and required storage destination for audit logs.** |

**🛠️ Implementation Path Chosen**

TechRisk defines **two supported approaches** for implementing PII access logging:

1. **Centralized Logging via Kafka** **(Deprecated)**
2. **Local Audit Logging**

**✅ 1. Local Audit Logging (Recommended & Supported)**

* **Current standard approach** as per updated TechRisk guidelines.
* Logs are generated locally in a **structured JSON format**, excluding sensitive values.
* Logs are written to a **secure local file or directly stored in BQL**

**Key Functional Requirements**

| **Feature** | **Description** |
| --- | --- |
| **PII Detection** | Detect fields in Protobuf messages annotated with isPii = true. |
| **Audit Logging** | Log each access of PII fields with metadata (e.g., user, endpoint, timestamp). |
| **Separate Log File** | Write PII audit logs to a dedicated file (e.g., pii-audit.log). |
| **Minimal Intrusion** | Should work through an interceptor without modifying existing service logic. |
| **Enable/Disable Toggle** | Ability to turn auditing on/off via configuration. |

.