TASK-4

Big Data Security and Compliance Framework

Case Study: Online Retail Dataset in Databricks (Unity Catalog)

1. Introduction

This report presents a security and compliance framework for a Big Data system implemented using Databricks (Unity Catalog). The system stores and processes an Online Retail dataset containing transactional data including:

- InvoiceNo
- StockCode
- Description
- Quantity
- InvoiceDate
- UnitPrice
- CustomerID
- Country

Since CustomerID can be linked to individuals, the dataset falls under GDPR scope (personal data).

The objective is to design a framework ensuring:

- Data confidentiality
- Data integrity
- Data availability
- Regulatory compliance (GDPR principles)

2. System Architecture

The implemented big data pipeline consists of:

2.1 Data Ingestion Layer

- Online Retail dataset uploaded to Databricks
- Stored inside Unity Catalog

2.2 Storage Layer

- Managed tables in Unity Catalog
- Backed by encrypted cloud storage

2.3 Processing Layer

- Data cleaning
- Filtering
- Grouping
- Aggregation using Spark

2.4 Analytics Layer

- Dashboard created using aggregated datasets

3. Data Classification

Data Field	**Classification**
InvoiceNo	Business Data
StockCode	Product Data
Description	Product Data
Quantity	Transaction Data
InvoiceDate	Transaction Data
UnitPrice	Financial Data
CustomerID	**Personal Data (GDPR – Pseudonymous)**
Country	Indirect Personal Data

CustomerID is considered **pseudonymous personal data**, as it can identify individuals when combined with other data.

4. Security Framework Design

The framework is divided into 5 control domains.

4.1 Access Control (Authorization)

Implemented using Unity Catalog Role-Based Access Control (RBAC):

Controls:

1. Grant SELECT only to authorized groups
2. Restrict raw table access
3. Provide access only to aggregated views for dashboard users
4. Use least privilege principle

Example Policy:

- Raw table → Accessible only to data_engineer
- Aggregated view → Accessible to data_analyst
- Dashboard users → No access to CustomerID

This ensures **Data Minimization (GDPR Principle)**.

4.2 Data Protection

4.2.1 Encryption at Rest

- Data stored in encrypted cloud storage
- Protects against storage theft or disk compromise

4.2.2 Encryption in Transit

- HTTPS/TLS used for:
  - Browser to Databricks
  - Workspace to storage
- Prevents interception attacks

4.3 Data Masking & Anonymization

To reduce compliance risk:

- Remove CustomerID from dashboard layer
- Create masked views:

Example:

```
SELECT Country, SUM(UnitPrice * Quantity) AS TotalRevenue
FROM retail_table
GROUP BY Country;
```

No personal identifiers exposed.

Optional:

- Hash CustomerID if needed
- Tokenization for advanced compliance

4.4 Auditing & Monitoring

Unity Catalog enables:

- Audit logs

- Query history tracking
- Access monitoring

Controls:

- Log who accessed CustomerID
- Log data modifications
- Alert on unauthorized access

This ensures **Accountability (GDPR Article 5)**.

4.5 Data Governance

4.5.1 Data Retention Policy

- Define retention period (e.g., 5 years)
- Automatic deletion after expiration

4.5.2 Right to Erasure (GDPR)

- Ability to delete CustomerID records
- Use DELETE queries with tracking

4.5.3 Data Lineage

- Track data from raw table → processed table → dashboard
- Maintain documentation

5. Compliance Mapping (GDPR)

GDPR Principle	Implementation
Lawfulness	Controlled access to personal data
Data Minimization	Dashboard only uses aggregated data
Integrity & Confidentiality	Encryption + RBAC
Accountability	Audit logs
Storage Limitation	Retention policy

6. Risk Assessment

Risk	**Mitigation**
Unauthorized table access	RBAC
Data breach	Encryption
Insider misuse	Audit logs
Excessive data exposure	Aggregated views

7. Prototype Demonstration

The prototype demonstrates secure handling by:

1. Storing dataset in Unity Catalog
2. Restricting raw table access
3. Creating aggregated views without CustomerID
4. Generating dashboard from sanitized data
5. Enabling audit tracking

8. Conclusion

This framework ensures that the Online Retail Big Data system:

- Protects personal data (CustomerID)
- Implements encryption at rest and in transit
- Enforces least privilege access
- Maintains auditability
- Complies with GDPR security principles

The system demonstrates practical enterprise-level security controls within Databricks.