Data Leakage

ML:

Scenarios	Mitigation Steps
Data leakage during training	**Ensure data separation between test and train data.** **Validate data pipelines and Review feature engineering:** Implement data quality checks and validation procedures to ensure that the data being used for training is accurate and free from anomalies. Review feature engineering processes to ensure that no information from the target variable is introduced into the training features. **Employ cross-validation techniques:** Use cross-validation techniques, such as k-fold cross-validation, to evaluate the model's performance on unseen data and prevent overfitting. - Monitor the model's performance over time to detect any signs of data leakage or performance degradation.
Data leakage during deployment	**Secure data storage and access:** Implement robust data security measures to protect both training and production data from unauthorized access or modification. **Monitor data pipelines:** Monitor data pipelines and production environments to detect any or unauthorized data access. Use DLP tools to identify and prevent sensitive data from being accidentally exposed or leaked during production operations.
Data leakage through external sources	**Implement data anonymization:** Anonymize sensitive data before using it for training or deployment to protect individual privacy . **Establish data sharing agreements:** When sharing data with external parties, establish clear data sharing agreements that outline usage restrictions and data protection measures.

IOT & Security:

Data leakage due to vulnerabilities in IoT devices that collect data for ML models	**Implement robust security measures on IoT devices:** Regularly update firmware and software on IoT devices to address security vulnerabilities. **Encrypt data transmission between IoT devices and cloud platforms:** Use encryption protocols, such as TLS/SSL, to safeguard data transmission between IoT devices and other endpoints. **- Monitor device activity for anomalies:** Establish a monitoring system to track IoT device activity, identify anomalies, and promptly address suspicious behavior. **Securely store data collected by IoT devices:** Store sensitive data collected by IoT devices in secure locations, such as encrypted databases or cloud storage with robust access controls**.**
Data leakage during device configuration and setup	**Provide clear user manuals and instructions:** Provide clear and comprehensive user manuals and instructions for IoT device configuration and setup, emphasizing secure practices. **- Implement default security settings:** Set strong default passwords and enable security features for IoT devices upon initial setup to reduce the risk of unauthorized access. **Educate users on secure device configuration:** Educate users about the importance of secure device configuration and provide guidance on selecting strong passwords, enabling encryption, and configuring security settings. **Offer secure remote configuration options:** Implement secure remote configuration options for IoT devices, allowing authorized personnel to manage device settings without physical access.
Unauthorized Access to ML Model or Training Data	**Implement strong authentication for access to ML models and datasets.** **Encrypt ML model files and training datasets at rest and in transit.** **Utilize access controls to restrict permissions based on roles.** **Monitor access logs for suspicious activities and anomalous behavior.**
Lack of data access controls:	**Implement data access controls**: Implement granular data access controls, granting access to sensitive data only to authorized individuals and systems. Use the principle of least privilege, granting access only to the data needed for specific tasks.

Insecure network connections:	Securely store IoT devices in restricted areas and implement access controls to prevent unauthorized physical access. Consider using tamper-evident seals to detect unauthorized device tampering
Insecure IOT device configurations:	**Change default passwords:** Upon receiving an IoT device, immediately change its default password to a strong, unique one. This simple step can significantly enhance security. **Implement strong authentication mechanisms for device access**
Insecure Communication Between IoT Devices and ML Models	**Encrypt data exchanged between IoT devices and ML models using secure protocols (e.g., TLS).** **Implement secure APIs for communication, with proper authentication and authorization mechanisms.** **Use VPNs to secure data transmission between IoT devices and ML models, especially in remote settings.** **Regularly update and patch communication protocols to address vulnerabilities.**