# **Proposal for Implementing Zero Trust Security in a Forgery Detection Web Application**

## **1. Introduction**

This proposal outlines the implementation of Zero Trust Security (ZTS) in a web application that verifies whether a document is forged or authentic. Given the sensitive nature of document verification, adopting Zero Trust principles will enhance security by ensuring continuous authentication, strict access control, encrypted data handling, and real-time monitoring.

## **2. Objectives**

* Prevent unauthorized access to document verification services.
* Ensure the confidentiality and integrity of uploaded documents.
* Secure API communications and prevent adversarial attacks.
* Implement continuous monitoring and logging to detect anomalies.

## **3. Zero Trust Security Implementation Plan**

### **3.1 User Authentication & Access Control**

* **Multi-Factor Authentication (MFA):** Require users to authenticate via OTP, email verification, or biometric authentication.
* **Role-Based Access Control (RBAC):**
  + Users: Can upload and check documents.
  + Admins: Can review flagged documents and manage users.
* **Least Privilege Access:** Restrict permissions to only necessary functionalities.

### **3.2 Document Security & Encryption**

* **Encryption:** Store all uploaded documents in an encrypted format (AES-256) to prevent unauthorized access.
* **Hashing for Integrity:** Generate a unique SHA-256 hash for each document and verify it during processing to detect tampering.
* **Watermarking for Tamper Detection:** Embed invisible digital watermarks in scanned images to prevent forgery attempts.

### **3.3 API & Model Security**

* **JWT Authentication for API Calls:** Require users to authenticate before interacting with the verification API.
* **Rate Limiting & Throttling:** Use API Gateway, NGINX, or Cloudflare to prevent excessive API requests and DoS attacks.
* **Input Validation & Sanitization:** Filter and sanitize document inputs before processing them to mitigate injection attacks.

### **3.4 Network & Deployment Security**

* **Zero Trust Network Access (ZTNA):**
  + Host the verification model and database on private cloud endpoints with restricted access.
* **Microsegmentation:**
  + Use containerized deployment (Docker/Kubernetes) to isolate different services (authentication, processing, and storage).
* **Logging & Intrusion Detection:**
  + Monitor logs using SIEM tools (Splunk, ELK Stack, AWS CloudTrail) to detect anomalies and unauthorized access.

## **4. Expected Outcomes**

* Enhanced security and access control for document verification services.
* Protection against unauthorized access, tampering, and data breaches.
* Secure model deployment with minimal risk of adversarial attacks.
* Continuous monitoring for proactive threat detection and mitigation.

## **5. Conclusion**

Implementing Zero Trust Security in the forgery detection web application will ensure a high level of security, data integrity, and user privacy. The proposed measures will safeguard sensitive documents and protect against emerging cyber threats.