Security and HIPAA Compliance Plan – AI on FHIR

My understanding of HIPAA (Health Insurance Portability and Accountability Act) is that it's a U.S. law designed to protect patients' health data. It emphasizes keeping health information private, secure, and accessible only to the right people. HIPAA especially matters when dealing with PHI (Protected Health Information), and in this project—where we're working with simulated FHIR data—I've tried to think through how a real system could be built to align with these principles.

1. Authentication and Authorization

To ensure that only authorized users can access patient data, I would implement OAuth 2.0 as the authentication and authorization protocol. Since this is a healthcare-focused application using FHIR APIs, I would extend OAuth with SMART on FHIR, which provides a standardized way to define what data the application can access.

This setup helps ensure that - Users sign in securely, Tokens define the level of access, Frontends and APIs only allow data access based on permissions. In public clients like a React frontend, I'd also include PKCE (Proof Key for Code Exchange) to make the OAuth flow more secure.

2. Data Privacy and Audit Logging

HIPAA's main goal is to protect sensitive health data. So in my implementation - All API calls will use HTTPS to encrypt data in transit. Any stored data will be encrypted using AES-256 encryption. Sensitive fields would be minimized or masked unless strictly needed. Every access to a patient record (read or write) will be logged with - who accessed it, what they accessed, and when.

This audit trail helps ensure accountability and gives administrators a way to trace any suspicious activity. For machine learning or analytics use cases, I'd make sure that PHI is de-identified before being processed. I learned that HIPAA allows either Safe Harbor or Expert Determination methods for anonymization.

3. Role-Based Access Control (RBAC)

Since different types of users need different levels of access, I would enforce role-based access control. For example - Admins would have full access to manage users and data. Clinicians could view and edit patient data relevant to them. Researchers would only see de-identified patient data for analysis. Access control rules can be enforced either through middleware or token scopes, and checked at the API level before allowing data access.

Summary

To summarize, my approach to HIPAA compliance in this project includes:

- Using OAuth 2.0 with SMART on FHIR for secure, standards-based authentication,
- Encrypting data both in transit and at rest,
- Logging all actions for traceability, and
- Enforcing strict role-based access control to make sure people only access what they're allowed to.

Even though the FHIR data here is simulated, I tried to approach the project with real-world privacy and security in mind. This experience gave me a deeper appreciation of how sensitive health data is and how seriously it needs to be protected in production systems.