Evaluation of Coupling and Cohesion:

Coupling: The current authentication module demonstrates a high level of coupling, wherein changes in one component often trigger modifications in closely connected components. For instance, alterations to the database schema necessitate corresponding updates in the authentication logic.

Cohesion: Cohesion within the authentication module is moderately structured, with related functionalities grouped together. However, there exists an opportunity for improvement to ensure that each module or component concentrates on a single responsibility, enhancing overall cohesion.

Proposed Modifications:

Decoupling from Data Access Layer:
Develop a distinct authentication service layer tasked with managing interactions with the data access layer. This separation diminishes coupling between authentication logic and database operations.

Introduction of Dependency Injection:
Implement dependency injection techniques to inject dependencies into the authentication module. This approach facilitates the substitution of components and mitigates tight coupling.

Refactoring Business Logic:
Restructure the authentication logic by segregating concerns like user validation, password hashing, and session management into discrete classes or modules. This refactoring promotes improved cohesion and modularity.

Standardization of Interfaces:
Establish standardized interfaces for authentication-related functionalities, including user authentication, password reset, and user profile management. Standardization fosters interoperability and simplifies integration with external systems or future enhancements.

Implementation of Pub/Sub Model:
Introduce a publish-subscribe model to manage authentication events, allowing various components to subscribe to pertinent events and react accordingly. This approach minimizes tight coupling between components, enhancing system flexibility and maintainability.