

Scaling for Production: Key Points

- **Scalability & Performance:** To handle thousands of customers and their file processing needs daily, the system must be scalable. This means it can grow to handle more work as needed. Using AWS services like Lambda, S3, and SQS allows us to automatically scale up or down based on the demand.
- **Reliability:** The system must be reliable, ensuring it's always available for users and that file processing is completed successfully without data loss. Services like S3 provide durable storage, while using multiple Lambda functions or ECS tasks can ensure processing continues even if one instance fails.
- **Security:** Keeping user data and files secure is crucial. Using Cognito for authentication ensures that only authorized users can upload or access files. S3 and API Gateway also offer built-in security features to protect data in transit and at rest.

Constraints

- **Processing Time:** Large or complex EDF files might take longer to process, which could lead to delays. Lambda functions have a maximum execution time limit, which might not be sufficient for all processing needs.
- **Resource Limits:** AWS services have limits, though many can be increased upon request. Monitoring these limits and planning for increases is important as the application scales.
- **Cost:** Scaling to handle thousands of users will incur higher costs. It's important to monitor usage and costs closely, using AWS's pricing calculator and cost management tools.

Proposed Architecture

- **S3:** Stores all EDF files. It's highly scalable and integrates well with other AWS services for processing and analysis.
- **Lambda:** Processes files upon upload to S3. For files that require longer processing times than Lambda allows, trigger an ECS or EKS task.
- **SQS:** Manages processing jobs, ensuring that files are processed in an orderly manner and enabling the system to scale processing across multiple instances.
- **DynamoDB:** Keeps track of file processing status and other metadata, offering fast, scalable, no-SQL database solutions.
- **API Gateway + Cognito:** Manages file uploads and user authentication, ensuring secure access to the application.
- **ECS/EKS (Optional):** For heavy processing needs that exceed Lambda's capabilities, use container services for scalability and flexibility.
- **CloudWatch:** Monitors the application, providing logs, metrics, and alarms to ensure the system's health and performance.