AWS Provisioning Flow:

Device boots → Presents serial/cert/code

↓

Technician → Enters into portal (React + API Gateway)

↓

Lambda Function → Verifies identity against internal records (DynamoDB)

↓

Fleet Provisioning by Claim or Template → AWS IoT Core issues cert and registers device

↓

Device connects securely to AWS IoT Core using MQTT

Reporting and Auditing Logs:

Device → AWS IoT Core → Rules Engine

↓

Lambda Function stores

→ Device metadata → DynamoDB

→ Audit logs → RDS (User actions, updates, errors)

↓

User requests report → API Gateway + Lambda

↓

Lambda creates CSV/JSON → Uploads to S3 → Returns download URL

Role-Based Access Control (RBAC):

User logs in (Cognito)

↓

JWT contains custom claims like role

↓

Frontend and Lambda check role before permitting action

OTA Firmware Update System:

Admin uploads firmware file via UI

↓

Lambda stores binary in S3 and metadata in DB

↓

Admin selects devices or groups (by tag)

↓

Lambda creates an AWS IoT Job → targets selected devices

↓

Devices receive job document (with download URL)

↓

Devices download firmware from S3 and apply update

↓

Devices report update status → stored in DB

↓

Frontend shows: Queued / In Progress / Completed / Failed

WorkFlow Steps:

Device Auto-Onboarding Workflow:

1. Device boots up with identity info
   Device starts with a unique serial number or preloaded certificate.

2. Technician enters code into portal
   A technician inputs the device serial or code into a web app to initiate onboarding.
3. Backend verifies device authenticity
   A Lambda function checks the device info against an internal DB (ex: DynamoDB) to ensure it's valid and unregistered.
4. AWS IoT Fleet Provisioning registers the device
   The system uses IoT Core's Fleet Provisioning to create a unique certificate and register the device.
5. Device connects to AWS IoT Core using credentials
   The device securely connects to AWS IoT Core over MQTT with its new certificate and starts sending telemetry.

Role-Based Access Control (RBAC):

1. User logs in via Cognito
   Admin, Technician, or Viewer signs in using Amazon Cognito and receives a JWT with their role in token claims.
2. Token sent with API requests
   All frontend-to-backend communication includes this token for role-based enforcement.
3. Backend checks role before action
   Lambda or API Gateway inspects the token to allow or deny actions like firmware updates or report downloads.
4. Frontend hides unauthorized actions
   The UI conditionally shows or hides buttons and actions based on the user's role.

Reporting and Audit Logging:

1. Telemetry data stored in DynamoDB / Timestream
   Real-time data from devices (heartbeat, health, firmware info) is written to time-series DBs for reporting.
2. Audit events logged in RDS
   Actions like updates, login attempts, firmware pushes, or failures are logged in a structured SQL DB.

3. User requests report download
   A user calls an API (e.g., /generate-report?format=csv), choosing format and filters.
4. Backend compiles data and stores in S3
   Lambda queries data, generates CSV/JSON, uploads to S3, and returns a signed URL.
5. Audit logs viewable per user or device
   Logs can be filtered by user, device ID, or action type for accountability and debugging.

Over-The-Air (OTA) Firmware Update Workflow:

1. Admin uploads firmware via UI
   Admin uploads a new firmware binary to the portal, which is stored in S3 and versioned.
2. Firmware metadata saved in DB
   Metadata like version number, hash, upload date, and description is saved in RDS or DynamoDB.
3. Admin selects devices or groups for update
   Devices can be targeted individually or by group/tag through a dropdown or search.
4. Backend creates an IoT Job
   A Lambda function triggers an AWS IoT Job that includes the firmware download link and instructions.
5. Device receives job and downloads firmware
   The device receives the job, verifies the version and checksum, and downloads firmware from a secure S3 link.
6. Device installs update and reboots
   After validating the binary, the device applies the update and restarts.
7. Device reports update status
   The device sends back its status (In Progress, Completed, Failed) which is logged and shown in the UI dashboard.

Summary of the Complete Platform Flow:

- Devices are automatically registered only after verification.
- Users are securely authenticated and restricted by role.
- The system maintains a full audit trail and downloadable reports.
- Admins manage OTA updates in bulk with real-time progress tracking.