```
// Dependencies
const express = require('express');
const jwt = require('jsonwebtoken');
const jose = require('jose');
const bodyParser = require('body-parser');
const multer = require('multer');
const fs = require('fs');
const path = require('path');

// Constants
const JWT_SECRET = 'your-jwt-secret';
const JWE_SECRET = new TextEncoder().encode('your-jwe-encryption-secret'); // 256-bit
const upload = multer({ dest: 'uploads/' });

const app = express();
app.use(bodyParser.json());

// Mock User DB
const users = {
  user1: { id: 'user1', name: 'Alice', password: 'password123' },
};

// Auth - Login to get JWT
app.post('/auth/login', (req, res) => {
  const { username, password } = req.body;
  const user = users[username];
  if (!user || user.password !== password) return res.status(401).json({ message: 'Invalid
credentials' });

  const token = jwt.sign({ id: user.id, name: user.name }, JWT_SECRET, { expiresIn: '1h' });
  res.json({ token });
});

// Middleware - Auth using JWT
function authenticateJWT(req, res, next) {
  const authHeader = req.headers.authorization;
  if (!authHeader) return res.sendStatus(401);

  const token = authHeader.split(' ')[1];
  jwt.verify(token, JWT_SECRET, (err, user) => {
    if (err) return res.sendStatus(403);
    req.user = user;
    next();
  });
}

// Generate JWE for upload (restricted token)
app.post('/aadhaar/token', authenticateJWT, async (req, res) => {
  const payload = {
    upload: 'aadhaar',
    userId: req.user.id,
    masked: true,
    exp: Math.floor(Date.now() / 1000) + (5 * 60), // 5 minutes expiry
  };

  const jwe = await new jose.EncryptJWT(payload)
    .setProtectedHeader({ alg: 'dir', enc: 'A256GCM' })
    .setIssuedAt()
    .setExpirationTime('5m')
    .encrypt(JWE_SECRET);

  res.json({ jwe });
});

// Aadhaar Upload (validate JWT + JWE)
app.post('/aadhaar/upload', authenticateJWT, upload.single('aadhaar'), async (req, res) => {
  const jweToken = req.headers['x-jwe-token'];
  if (!jweToken) return res.status(400).json({ message: 'Missing JWE token' });

  try {
    const { payload } = await jose.jwtDecrypt(jweToken, JWE_SECRET);
    if (payload.userId !== req.user.id || payload.upload !== 'aadhaar') {
      return res.status(403).json({ message: 'Invalid token scope' });
    }

    return res.json({ message: 'Masked Aadhaar uploaded successfully!', file: req.file });
  } catch (err) {
    return res.status(403).json({ message: 'Invalid or expired JWE token' });
  }
});

// Start server
app.listen(3000, () => {
  console.log('Server running on http://localhost:3000');
});
```