# Secure File Transfer Workflow (AES-256-GCM + RBAC)

## 📤 Upload Workflow

1. 1. User logs in (session created with user\_id and role).
2. 2. User uploads a file via dashboard (upload.php).
3. 3. File is read into PHP.
4. 4. File contents encrypted with AES-256-GCM (random IV, tag generated).
5. 5. Encrypted file (.bin) saved in uploads/ directory.
6. 6. Metadata stored in MySQL (original filename, encrypted filename, iv, tag, user\_id).
7. 7. User redirected back to dashboard showing their uploaded file.

## 📥 Download Workflow

1. 1. User clicks download link (download.php?id=X).
2. 2. PHP checks role-based permissions (admin can download all, user only their own).
3. 3. Metadata (original filename, encrypted filename, iv, tag) fetched from MySQL.
4. 4. Encrypted file (.bin) loaded from uploads/ folder.
5. 5. PHP decrypts file using AES-256-GCM (same key, iv, tag).
6. 6. Decrypted plaintext streamed to browser with original filename.

## 🔐 Role-Based Access Control (RBAC)

* Users can upload and download only their own files.
* Admins can view/download all files and manage users.
* RBAC enforced in dashboard.php (UI) and download.php (permissions check).

## 📑 Summary Workflow

Upload: File → PHP → Encrypt (AES-256-GCM) → Save ciphertext in uploads/ → Metadata in DB  
Download: Metadata → Load ciphertext → Decrypt (AES-256-GCM) → Serve plaintext to browser