Kyocera Scan to mail via Oauth2

Option 1: Use Kyocera Exchange Online Connector (Recommended for older devices)

Kyocera provides a free utility called **Kyocera Exchange Online Connector**, which enables OAuth 2.0 authentication without requiring firmware upgrades. It acts as a middleware between your MFP and Microsoft 365.

Steps:

1. **Download the Connector**
   Get it from Kyocera's official site. [kyoceradoc...lutions.us]
2. **Install on a Windows Server or PC**
   - Minimum: 4-core CPU, 8 GB RAM.
   - Supported OS: Windows or Windows Server.
3. **Configure Microsoft 365**
   - Log in to **Microsoft 365 Admin Center**.
   - Register the connector as an **Azure AD App** for OAuth.
   - Grant **SMTP.Send** permissions and consent.
4. **Configure the Connector**
   - Enter your Microsoft 365 tenant details.
   - Authenticate using OAuth (you'll get a token instead of username/password).
5. **Point Kyocera MFP to the Connector**
   - In the device's **Command Center RX**, set SMTP server to the connector's IP.
   - Use port 587 with STARTTLS.

Option 2 Direct OAuth 2.0 on Kyocera (Newer models with firmware support)

1. **Register a Single App in Azure AD**
   - Go to **Azure Portal** → **App Registrations** → **New Registration**.
   - Name it something like Kyocera-MFP-OAuth.
   - Set **Redirect URI** to a generic placeholder (e.g., https://localhost or one device's callback URL).
     Tip: Kyocera devices typically don't require the redirect URI for SMTP OAuth because they use the **client credentials flow**.

2. **Assign Permissions**
    - Under **API Permissions**, add:
        - SMTP.Send (for sending email).
    - Grant **Admin Consent** for the tenant.
3. **Create Client Secret**
    - Copy the **Client ID**, **Client Secret**, and **Tenant ID**.
4. **Configure Each Kyocera Device**
    - In **Command Center RX → Email Settings**:
        - SMTP Server: smtp.office365.com
        - Port: 587
        - Security: STARTTLS
        - Authentication: OAuth 2.0
        - Enter **Client ID**, **Client Secret**, and **Tenant ID** from the single app.
5. **Token Handling**
    - Each device will request its own OAuth token using the shared app credentials.
    - Microsoft allows multiple concurrent tokens for the same app.