Outlook Calendar Integration

Roy OS Integration Guide | **READ + WRITE** **USER-DELEGATED**

Overview

The Roy OS Outlook Calendar integration reads your calendar to prepare meeting context (who's attending, what's on the agenda, which room) and creates follow-up calendar events from meeting action items. Access is user-delegated — Roy only sees your own calendar, and each person consents individually.

What Roy Can Do with Your Calendar

Capability	Description
Read your meetings	Reads meeting titles, times, attendees, room assignments, and descriptions to prepare contextual briefings before meetings.
Create follow-ups	Creates follow-up calendar events (review sessions, check-ins) from meeting action items. Events appear on your calendar only.
Update events	Updates event details (time, description) for follow-ups the agent created.

> **What Roy cannot do:** Read your email, send messages on your behalf, access other people's calendars, browse the organization directory, or perform any admin operations. Roy only accesses your personal calendar view.

Required Permissions

Microsoft Graph Scope	Why It's Needed
`Calendars.ReadWrite` (delegated)	Read your calendar events and create follow-up events on your behalf

This is a **delegated** permission — each user must individually consent. There is no application-level or admin-consented calendar access.

Setup Guide

Prerequisites

You'll need access to your organization's Microsoft Entra ID (Azure AD) tenant to create an app registration, and access to your Azure Key Vault.

1 **Create an App Registration**

In Entra ID → **App registrations** → **New registration**. Name it "Roy OS". Set the redirect URI provided by Roy AI during onboarding. Select "Accounts in this organizational directory only."

2 **Configure API Permissions**

Under **API permissions** → **Add a permission** → **Microsoft Graph** → **Delegated permissions** → add `Calendars.ReadWrite`. Do not add application permissions or admin-consent scopes.

3 **Generate a Client Secret**

Under **Certificates & secrets** → **New client secret**. Set expiry per your organization's policy (Entra ID default is 90 days). Copy the secret value.

4 **Store Credentials in Key Vault**

Secret Name	Value
`graph-client-id`	Application (client) ID from the app registration
`graph-client-secret`	Client secret value

5 **User Consent**

No admin action required beyond the app registration. Each user sees the Azure consent screen on first use and individually approves calendar access. Users can revoke at any time.

Managing Access

Action	How
Revoke your own access	Go to myapps.microsoft.com → find Roy OS → Revoke. Only your calendar access is removed.
Revoke all users	Entra ID admin deletes or disables the app registration. All delegated tokens invalidated.
Limit scope	

	Entra ID admin can restrict the app's allowed permissions via the app registration settings.
Rotate credentials	Generate a new client secret in Entra ID. Update the Key Vault secret. Old secret expires per your policy.

FAQ

Can Roy read my email?

No. Roy only requests `Calendars.ReadWrite`. It has no mail, contacts, or files permissions.

Can my manager see my calendar through Roy?

No. Roy accesses each user's calendar individually via their own delegated token. One person's calendar data is never shared with another through Roy.

What follow-up events look like?

Roy creates standard Outlook calendar events with a title like "Follow-up: [Action Item]", a description referencing the original meeting, and attendees if applicable. You can edit or delete these events like any other calendar event.

> **Audit trail:** Every Graph API call is logged (action type, event ID, timestamp, success/failure) and available in your Azure Log Analytics workspace.