LAB 365 TECHNICAL PLAYBOOK · FIRST EDITION (2026)

AI for Microsoft 365 Administrators

The Practical Copilot, Autonomous Agents, Purview DSPM & Zero Trust Security Playbook — Built with PowerShell 7, Microsoft Graph SDK, and Production Runbooks.

Semantic Index & Graph | SharePoint SAM & DAG | Purview DSPM & DLP

Entra Agent ID & MXC 1.0 | Security Copilot & KQL | PAYG Cost Caps & ROI

Sunil Chauhan
Enterprise Messaging, Cloud Identity & AI Governance · www.lab365.in

Complete 10-Chapter Edition
November 2026

About This Playbook

AI for Microsoft 365 Administrators is a hands-on engineering handbook written for Microsoft 365 administrators, Exchange/SharePoint architects, and cloud security engineers tasked with deploying and governing Microsoft 365 Copilot, Copilot Studio agents, and Security Copilot without exposing sensitive enterprise data.

Every chapter moves directly from architectural mechanics (Semantic Index, Graph grounding, Entra Agent Identity Blueprints, Microsoft Execution Containers) into copy-pasteable PowerShell 7, Microsoft Graph SDK, and KQL runbooks validated against a live Microsoft 365 tenant.

What You Will Master Across 10 Chapters

- **Chapters 1–3 (Data Readiness & SharePoint):** Semantic Index grounding, fixing SharePoint EEEU oversharing, Restricted SharePoint Search, and SharePoint Advanced Management (SAM / DAG / RAC).
- **Chapters 4–5 (Microsoft Purview & Shadow AI):** Copilot DLP policies scoped by Entra Administrative Units, DSPM for AI, Endpoint DLP, Defender local agent discovery, and Intune Edge controls.
- **Chapters 6–8 (Autonomous Agents & Copilot Studio):** Entra Agent ID, Agent 365, Microsoft Execution Containers (MXC 1.0), Power Platform Managed Environments, and Connector Write Actions.
- **Chapters 9–10 (SecOps & FinOps):** Security Copilot + Sentinel KQL threat hunting, Pay-As-You-Go (PAYG) billing caps, `CopilotInteraction` UAL auditing, and license reclamation ROI.

Prerequisites & Recommended Tooling

- **PowerShell 7.4+** with `Microsoft.Graph`, `ExchangeOnlineManagement` (v3), and `Microsoft.Online.SharePoint.PowerShell`.
- A Microsoft 365 E3/E5 or Copilot-enabled tenant (or non-production dev tenant) with least-privilege PIM roles.

Author: Sunil Chauhan (linkedin.com/in/sunilkms) · Published by **LAB 365** (https://www.lab365.in).
© 2026 Sunil Chauhan / LAB 365. Free for individual enterprise administrators and internal IT teams. Please link back to `www.lab365.in` when sharing or citing scripts.

Table of Contents (10 Chapters)

Ch 01 **How Microsoft 365 Copilot Actually Works (Semantic Index, Graph Grounding & Tenant Boundary)**

Chapter 1 of the AI for Microsoft 365 Administrators book series. Understand how Microsoft 365 Copilot works under the hood: Large Language Models, the Semantic Index, Microsoft Graph grounding, Entra ID permission boundaries, and your first readiness audit script.

Ch 02 **Fixing Data Oversharing Before Copilot Rollout (SharePoint ACLs, EEEU & Restricted SharePoint Search)**

A comprehensive enterprise runbook for Microsoft 365 Copilot security and governance. Learn how to audit over-permissioned SharePoint sites, enforce Purview sensitivity labels, inspect CopilotInteraction audit logs, and automate posture checks with PowerShell.

Ch 03 **SharePoint Advanced Management (SAM): Data Access Governance (DAG), Restricted Access Control (RAC) & Stale Site Hygiene**

Chapter 3 of AI for Microsoft 365 Administrators. Master SharePoint Advanced Management (SAM) unlocked by Copilot: Data Access Governance (DAG) reports, Restricted Access Control (RAC), Restricted Content Discovery (RCD), Site Access Reviews, and Inactive Site Policies with PowerShell.

Ch 04 **Microsoft Purview for AI: Sensitivity Labels, Copilot DLP & Administrative Units**

How to build Microsoft Purview DLP policies that stop Microsoft 365 Copilot from processing sensitive content, and how to delegate them to regional admins with Entra administrative units using New-DlpCompliancePolicy and -PolicyRBACScopes.

Ch 05 **Purview DSPM for AI & Shadow AI: Detecting Unsanctioned AI Tools, Endpoint DLP & Intune Edge Controls**

Chapter 5 of AI for Microsoft 365 Administrators. Discover and govern Shadow AI (ChatGPT, Claude, Gemini, local coding agents like Copilot CLI and Claude Code) using Microsoft Purview DSPM for AI, Endpoint DLP, Defender local agent discovery, and Intune Microsoft Edge policies.

Ch 06 **Governing Autonomous AI Agents: Entra Agent ID, Blueprints & Agent 365**

How to inventory, govern, and apply Conditional Access to autonomous AI agents using Microsoft Entra Agent ID (Ignite 2026 Preview), Agent 365, and Microsoft Graph PowerShell.

Ch 07 **Sandboxing Local & Cloud AI Agents: Microsoft Execution Containers (MXC 1.0) & Dynamic Permissions**

What Microsoft Execution Containers (MXC) 1.0 does for AI agent security, how it fits with Entra Agent ID blueprints, and how to inventory agent identity blueprints with Microsoft Graph PowerShell.

Ch 08 **Copilot Studio, SharePoint Agents & Connector Write Actions: Power Platform & API Governance**

Chapter 8 of AI for Microsoft 365 Administrators. Govern custom Copilot Studio agents, SharePoint agents, and GA Connector Write Actions using Power Platform Managed Environments, DLP connector groups, environment routing, and Integrated Apps approval workflows.

Ch 09 **Microsoft Security Copilot & Sentinel: AI-Assisted Threat Hunting & KQL Incident Triage**

Chapter 9 of AI for Microsoft 365 Administrators. Use Microsoft Security Copilot, Microsoft Sentinel data lakes, and Defender XDR KQL queries to accelerate phishing triage, hunt for rogue local AI agents, and automate incident response with SCU cost controls.

Ch 10 **Copilot Cost Management (PAYG Billing & Caps), CopilotInteraction Auditing & ROI Analytics**

Chapter 10 (Final Chapter) of AI for Microsoft 365 Administrators. Master Microsoft 365 Copilot seat vs. Pay-As-You-Go (PAYG) billing architecture, Azure budget caps, Viva Insights Copilot Analytics, and automated CopilotInteraction license reclamation with PowerShell.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 01 OF 10

How Microsoft 365 Copilot Actually Works (Semantic Index, Graph Grounding & Tenant Boundary)

Chapter 1 of the AI for Microsoft 365 Administrators book series. Understand how Microsoft 365 Copilot works under the hood: Large Language Models, the Semantic Index, Microsoft Graph grounding, Entra ID permission boundaries, and your first readiness audit script.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch1-how-copilot-works-semantic-index-graph.html

📛 CHAPTER 1 · LEARNING OBJECTIVES AI FOR M365 ADMINS · BOOK SERIES

What You Will Master in Chapter 1

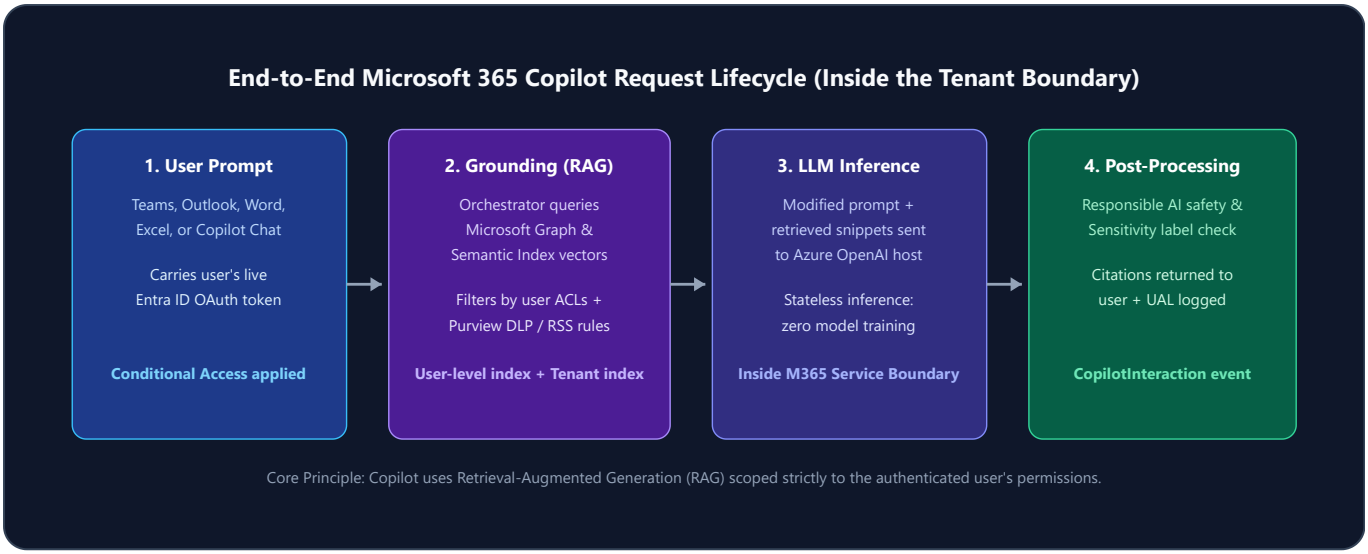
- **The 3-Engine Architecture:** How Microsoft 365 apps, Microsoft Graph + Semantic Index, and the LLM orchestrator work together during every prompt.
- **The Security Truth:** Why Copilot respects every existing SharePoint, OneDrive, Teams, and Exchange ACL—and why *oversharing* is the real threat.
- **Tenant Data Boundary:** What leaves your tenant (nothing for model training), what stays inside the Microsoft 365 service boundary, and how web grounding works.
- **Hands-On Lab:** Run a read-only Microsoft Graph PowerShell readiness audit against your tenant to inventory Copilot-eligible SKUs and baseline settings.

```
Get-MgSubscribedSku | Select SkuPartNumber, ConsumedUnits, @{N='Enabled';E=
{$_.PrepaidUnits.Enabled}}
```

Before you configure a single Data Loss Prevention policy or assign a single Microsoft 365 Copilot license, you need an accurate mental model of **what happens in the 3 seconds between a user hitting Enter and Copilot displaying an answer**.

Most executive misconceptions—both "*Copilot will train ChatGPT on our board decks*" (false) and "*Copilot won't show confidential files because they're in an HR folder*" (false if permissions are broken)—come from misunderstanding the underlying architecture.

In Chapter 1, we strip away the marketing diagrams and trace the exact technical request lifecycle inside Microsoft 365.



---

1. The Three Building Blocks Every M365 Admin Must Understand

A. Retrieval-Augmented Generation (RAG), Not Fine-Tuning

Microsoft does **not** train or fine-tune a custom AI model on your company's emails and SharePoint files. Instead, Copilot uses **Retrieval-Augmented Generation (RAG)**:

1. The user asks: "*Summarize the Q4 network migration risks from last week's emails and SharePoint docs.*"
2. The Copilot orchestrator translates that prompt into search queries against **Microsoft Graph** and the **Semantic Index**.
3. Only the relevant text chunks the user has **Read** permission to see are retrieved and injected into a temporary prompt context ("grounding").
4. The LLM generates a response from that temporary context, attaches clickable source citations, and immediately discards the working memory.

B. The Semantic Index (User-Level vs. Tenant-Level)

Traditional keyword search fails when a user asks about "*layoff risks*" while the document says "*headcount reduction plan*". The **Semantic Index** solves this by creating conceptual vector embeddings of your organization's content:

- **User-Level Index:** Covers the user's own Exchange Online mailbox, Teams chats, and personal OneDrive files. Stored inside the user's mailbox substrate boundary.
- **Tenant-Level Index:** Covers SharePoint Online sites and shared libraries. When a user prompts Copilot, the index only returns vectors for documents whose Access Control Lists (ACLs) include that user.

C. Web Grounding (Bing Search Toggle)

When **Allow web search in Copilot** is enabled, the orchestrator can generate a short, de-identified web search query (stripped of tenant IDs and user names) and send it to the Bing search service to fetch public internet facts (e.g., current CVE numbers or public vendor pricing). Admins can control this via Cloud Policy (`Allow web search in Copilot`) or the Microsoft 365 Admin Center.

---

2. The Myth vs. Reality Table (Share With Your Leadership)

Common Executive Concern	Technical Reality in Microsoft 365
"Will our prompts train public OpenAI models?"	**No.** Prompts, grounded data, and responses remain inside the Microsoft 365 service boundary and are never used to train foundation models.
"Can Copilot read an HR folder if the user isn't in HR?"	**Only if permissions are broken.** If the HR SharePoint site was shared with `Everyone except external users`, Copilot can read it—because the user already could.
"Does Copilot bypass sensitivity label encryption?"	**No.** If a document is encrypted with a Purview sensitivity label that does not grant the user `View` and `Extract` rights, Copilot cannot ground on it.
"Can we see what files Copilot referenced?"	**Yes.** Every interaction generates a `CopilotInteraction` record in the Microsoft Purview Unified Audit Log listing the user, app host, and accessed resources.

---

3. The Microsoft 365 Copilot Licensing Model Explained (Free vs. E3 vs. E5)

Before purchasing a single add-on seat, every administrator and CIO needs a crystal-clear understanding of **what is included for free**, **what the $30/user/month Microsoft 365 Copilot add-on unlocks**, and **the exact security gap between Microsoft 365 E3 and Microsoft 365 E5**.

Side-by-Side Capability & Governance Matrix

Capability / Governance Control	Free: Copilot Chat (Included in E3 / E5 / Business)	M365 E3 + Paid Copilot ($30/user/mo)	M365 E5 (or E3 + E5 Compliance) + Paid Copilot ($30/user/mo)
Web-Grounded Chat + Enterprise Data Protection (EDP)	**Included** (Prompts not used for training)	**Included**	**Included**
Work Grounding (Graph + Semantic Index across Mail, Teams, SharePoint)	**No** (Manual file upload only)	**Included**	**Included**
Embedded Copilot in Word, Excel, PowerPoint, Outlook & Teams	**No**	**Included**	**Included**
SharePoint Agents & Custom Copilot Studio Agents	**Pay-As-You-Go (PAYG)** Azure meter	**Included** (for M365 data)	**Included** (for M365 data)
SharePoint Advanced Management (SAM: DAG, RAC, RCD)	**No** (Requires standalone SAM add-on)	**Unlocked Free** (Included with Copilot license)	**Unlocked Free** (Included with Copilot license)
Manual Purview Sensitivity Labels	**Included**	**Included**	**Included**
Auto-Labeling & Default SharePoint Library Labels	**No**	**No** *(Requires E5 / E5 Compliance)*	**Included**
Purview DLP for Copilot (`ExcludeContentProcessing`)	**No**	**No** *(Requires E5 / E5 Compliance)*	**Included**
Endpoint DLP & Insider Risk for Shadow AI (ChatGPT/Claude)	**No**	**No** *(Requires E5 / E5 Compliance)*	**Included**
Unified Audit Log (`CopilotInteraction`)	**Included** (180-day standard)	**Included** (180-day standard)	**Included** (1-year+ Audit Premium)

Key Takeaways for M365 Administrators

1. **Free (`Microsoft 365 Copilot Chat`) is your Safe Harbor against Shadow AI:** Every user with an eligible base license (M365 E3, E5, Business Standard/Premium, Office 365 E3/E5, or F1/F3) already gets **Copilot Chat** at `$0` with **Enterprise Data Protection (EDP)**. Use it as the sanctioned alternative to consumer ChatGPT for employees who do not need a $30/month full seat—and optionally link Azure **Pay-As-You-Go (PAYG)** billing if you want them to query SharePoint Agents (covered in Chapter 10).
2. **M365 E3 + Paid Copilot ($30/user/mo) unlocks full productivity + SAM, but lacks automated Purview enforcement:** Adding the $30 Copilot seat to an E3 user unlocks full Graph/Semantic Index grounding across Office apps AND unlocks core **SharePoint Advanced Management (SAM)** governance controls (covered in Chapter 3). However, E3 **does not** include Purview **Auto-Labeling**, **Copilot DLP**, or **Endpoint DLP** for Shadow

AI—so E3 tenants must rely heavily on **SharePoint ACL hygiene, RSS, and SAM (RAC/RCD)** unless they add the **Purview E5 Compliance** add-on.

3. **M365 E5 + Paid Copilot ($30/user/mo) is the full Zero Trust AI stack:** E5 gives you every control covered in this book—automated sensitivity labeling, **Purview DLP for Copilot** (Chapter 4), **DSPM for AI + Endpoint DLP** (Chapter 5), **Entra ID P2** risk-based Conditional Access, and **Defender for Endpoint P2** local agent discovery.
4. **Separate Consumption Models to Watch:** Copilot Studio and SharePoint Agents accessed by unlicensed users are billed via prepaid message packs or **Azure PAYG** meters (Chapter 10), while **Microsoft Security Copilot** is licensed separately via hourly Azure **Security Compute Units (SCUs)** (Chapter 9).

---

4. Hands-On Lab 1: Run Your Tenant Copilot Readiness Baseline

Before moving to Chapter 2 (fixing SharePoint oversharing), run this read-only Microsoft Graph PowerShell script to inventory your tenant's organization metadata, verified domains, and current Microsoft 365 / Copilot license SKUs.

POWERSHELL

```powershell
<#
.SYNOPSIS
    Chapter1-CopilotReadinessAudit.ps1
    Read-only Microsoft Graph baseline check for AI for M365 Admins (Chapter 1).
#>

# Connect with read-only Organization permissions (Interactive or CBA)
Connect-MgGraph -Scopes "Organization.Read.All", "Directory.Read.All" -NoWelcome

Write-Host "=== 1. Tenant Identity & Boundary ===" -ForegroundColor Cyan
Get-MgOrganization | Select-Object DisplayName, Id, TenantType, CreatedDateTime | Format-Table -AutoSize

Write-Host "=== 2. License Inventory (Checking Base E3/E5 & Copilot SKUs) ===" -ForegroundColor Cyan
$skus = Get-MgSubscribedSku
$skus | Select-Object SkuPartNumber, CapabilityStatus, ConsumedUnits,
    @{Name = "Purchased"; Expression = { $_.PrepaidUnits.Enabled } },
    @{Name = "Available"; Expression = { $_.PrepaidUnits.Enabled - $_.ConsumedUnits } } |
    Sort-Object SkuPartNumber | Format-Table -AutoSize

# Highlight any Copilot SKUs found
$copilotSkus = $skus | Where-Object { $_.SkuPartNumber -match "COPILOT|M365_AI" }
if ($copilotSkus) {
    Write-Host "[+] Copilot SKUs detected in tenant: $($copilotSkus.SkuPartNumber -join ', ')" -
ForegroundColor Green
} else {
    Write-Host "[i] No paid Microsoft 365 Copilot add-on SKUs assigned yet (ideal time to complete pre-
rollout governance!)." -ForegroundColor Yellow
}
```

Live Lab Output (LAB 365 Dev Tenant):

TEXT

```
DisplayName Id                                   TenantType CreatedDateTime
─────────── ──                                   ────────── ──────────────
LAB 365     ab85c7a1-366d-44ba-acc6-924fa0950045 AAD        2023-03-30T13:47:39Z
```

```
SkuPartNumber     CapabilityStatus ConsumedUnits Purchased Available
_____________     ________________ _____________ _________ _________

DEVELOPERPACK_E5 Enabled                     25        25         0
FLOW_FREE        Enabled                      5     10000      9995
```

---

5. Chapter 1 Lab Exercise & Next Step

Your Action Item Before Chapter 2:

1. Run `Chapter1-CopilotReadinessAudit.ps1` in your own dev or production tenant.
2. Verify whether your users hold qualifying base licenses (Microsoft 365 E3/E5, Business Standard/Premium) and whether **Purview E5 Compliance** is available for sensitivity labels and Copilot DLP.
3. Check whether **Web Grounding** is enabled or disabled in **Microsoft 365 Admin Center → Copilot → Settings**.

Up Next in the Book:

- **Chapter 2:** *Fixing Data Oversharing Before Copilot Rollout (SharePoint Permissions, EEEU & Restricted SharePoint Search)* — or preview our standalone M365 Copilot Tenant Security & Restricted SharePoint Search Runbook and Purview Copilot DLP Guide right now!

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 02 OF 10

Fixing Data Oversharing Before Copilot Rollout (SharePoint ACLs, EEEU & Restricted SharePoint Search)

A comprehensive enterprise runbook for Microsoft 365 Copilot security and governance. Learn how to audit over-permissioned SharePoint sites, enforce Purview sensitivity labels, inspect CopilotInteraction audit logs, and automate posture checks with PowerShell.

Online Runbook: www.lab365.in/2026/11/m365-copilot-tenant-security-data-governance-powershell-runbook.html

🚨 AI EXECUTIVE BRIEF & TL;DR VERIFIED FOR M365 COPILOT & PURVIEW 2026

Key Architectural Governance Takeaways

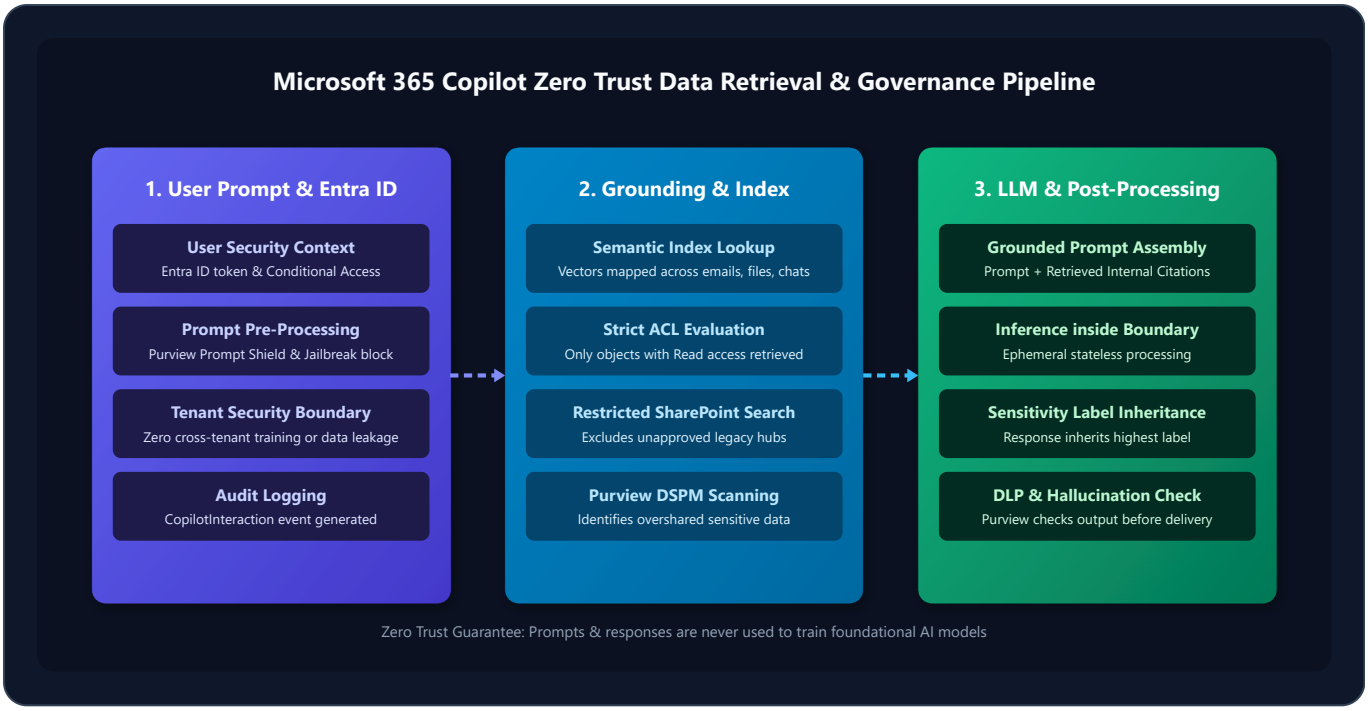
- **The Semantic Index respects user permissions:** Copilot never bypasses ACLs. If a user receives sensitive payroll or executive data in a response, it is because that data was overshared via `Everyone except external users` (EEEU) in SharePoint/OneDrive.
- **Purview Sensitivity Labels are mandatory:** Files encrypted with sensitivity labels that do not grant user `Extract` or `View` rights are automatically excluded from grounding and semantic indexing.
- **Audit Copilot prompts and responses:** Monitor Purview Unified Audit Log using the `CopilotInteraction` record type to track prompt metadata, grounded files, and user IDs without decrypting private user prompt text.

Connect-IPPSSession; Search-UnifiedAuditLog -RecordType CopilotInteraction -StartDate (Get-Date).AddDays(-1)

When enterprises roll out **Microsoft 365 Copilot**, security and messaging administrators quickly realize a fundamental truth: **Copilot does not cause data leaks; it exposes existing data oversharing.**

Because Copilot operates on behalf of the authenticated user using Microsoft Graph and the **Semantic Index**, it has access to every SharePoint site, Teams channel, OneDrive file, email, and meeting transcript that the user has permission to read. If a confidential financial spreadsheet was inadvertently shared with "*Everyone except external users*", Copilot will surface that data in employee prompts within seconds.

This engineering runbook details how to establish Zero Trust data security before and during a Copilot deployment, audit high-risk overshared repositories, enforce Microsoft Purview sensitivity labels, and monitor Copilot activity using PowerShell.



---

1. The Core Threat: Oversharing via Broad Group ACLs

The number one security issue in organizations enabling Copilot is **permission sprawl**:

- Users creating public Teams where confidential company strategy documents are stored.
- SharePoint document libraries where sharing links were set to `"Anyone with the link"` or `"People in your organization"`.
- Legacy intranet root sites granted to `Everyone except external users` (EEEU).

When an employee prompts Copilot:

> "*Summarize the upcoming executive compensation plan and reduction-in-force strategy,*"

If any document containing those keywords is accessible to `Everyone except external users`, Copilot reads the file, parses the data, and presents a detailed summary with direct citation links.

---

2. Restricting Copilot Scope via Restricted SharePoint Search (RSS)

If your organization has hundreds of legacy SharePoint sites and cannot audit every permission before launching Copilot, deploy **Restricted SharePoint Search (RSS)**:

Restricted SharePoint Search allows administrators to curate an **allowed list of up to 100 enterprise sites** that Copilot and enterprise search are permitted to reference. All other SharePoint sites are excluded from Copilot grounding unless the user has explicitly opened or modified the file recently.

Managing Restricted SharePoint Search via PowerShell

POWERSHELL

```powershell
# 1. Connect to SharePoint Online PowerShell
Connect-SPOService -Url "https://yourtenant-admin.sharepoint.com"

# 2. Check current Restricted SharePoint Search mode (Enabled or Disabled)
Get-SPOTenantRestrictedSearchMode

# 3. Enable Restricted SharePoint Search
Set-SPOTenantRestrictedSearchMode -Mode Enabled

# 4. Add Approved Enterprise Hub Sites to the Allowed List (up to 100 sites)
Add-SPOTenantRestrictedSearchAllowedList -SitesList @(
    "https://yourtenant.sharepoint.com/sites/Approved-Corporate-Policies",
    "https://yourtenant.sharepoint.com/sites/Employee-Handbook"
)

# 5. Verify Current Allowed List
Get-SPOTenantRestrictedSearchAllowedList
```

---

3. Auditing Overshared SharePoint Sites with PowerShell

To identify sites that have excessive broad access permissions, run this automated discovery script:

POWERSHELL

```powershell
<#
.SYNOPSIS
    Audit-CopilotOversharedSites.ps1
    Scans SharePoint Online sites for broad "Everyone except external users" permissions.
#>

Import-Module PnP.PowerShell -ErrorAction Stop

$TenantAdminUrl = "https://yourtenant-admin.sharepoint.com"
Connect-PnPOnline -Url $TenantAdminUrl -Interactive

$allSites = Get-PnPTenantSite | Where-Object { $_.Template -notlike "SRCHCEN*" -and $_.Template -notlike
"REDIRECT*" }
$findings = [System.Collections.Generic.List[PSCustomObject]]::new()

foreach ($site in $allSites) {
    Write-Host "Auditing: $($site.Url) ... " -ForegroundColor Cyan
    try {
        Connect-PnPOnline -Url $site.Url -Interactive
        $groups = Get-PnPGroup
        foreach ($group in $groups) {
            $users = Get-PnPGroupMember -Identity $group.Title -ErrorAction SilentlyContinue
            $hasEeeu = $users | Where-Object { $_.LoginName -like "*spo-grid-all-users*" -or $_.Title -
like "*Everyone except external users*" }
            if ($hasEeeu) {
                Write-Host "  [!] EXPOSURE FOUND: $($site.Title) grants access to Everyone except external
users!" -ForegroundColor Red
                $findings.Add([PSCustomObject]@{
                    SiteTitle = $site.Title
                    SiteUrl   = $site.Url
```

```
                    Group     = $group.Title
                    RiskLevel = "High (Copilot Grounding Exposure)"
                })
            }
        }
    } catch {
        Write-Warning "Could not inspect $($site.Url): $($_.Exception.Message)"
    }
}

$findings | Export-Csv -Path ".\Copilot-Overshared-Sites.csv" -NoTypeInformation
Write-Host "[+] Audit complete. Findings exported to Copilot-Overshared-Sites.csv" -ForegroundColor Green
```

---

4. Monitoring Copilot Interactions in Unified Audit Log (UAL)

Microsoft Purview logs every Copilot prompt interaction under the `CopilotInteraction` record type.

While Microsoft preserves user privacy by hashing or redacting the literal prompt content in standard audit logs, the log captures critical forensic metadata:

- The user who executed the prompt (`UserId`).
- The client application used (`Teams` , `Word` , `Excel` , `PowerPoint` , `Loop`).
- The specific files, emails, or meetings accessed and grounded (`AccessedResources`).
- The Purview sensitivity labels evaluated.

PowerShell Script: Extracting Copilot Interactions

POWERSHELL

```
# Connect to Exchange Online & Purview Compliance PowerShell
Connect-ExchangeOnline -ShowBanner:$false
Connect-IPPSSession -ShowBanner:$false

# Query Copilot Activity for the past 24 hours
$startDate = (Get-Date).AddDays(-1)
$endDate   = Get-Date

Write-Host "Querying Copilot audit events between $startDate and $endDate ..." -ForegroundColor Cyan

$auditLogs = Search-UnifiedAuditLog `
    -StartDate $startDate `
    -EndDate $endDate `
    -RecordType CopilotInteraction `
    -ResultSize 100

$results = foreach ($log in $auditLogs) {
    $auditData = $log.AuditData | ConvertFrom-Json
    [PSCustomObject]@{
        Timestamp    = $log.CreationDate
        User         = $log.UserIds
        Operation    = $log.Operations
        AppHost      = $auditData.AppHost
        GroundedFiles = ($auditData.AccessedResources | Select-Object -ExpandProperty Name) -join "; "
        Sensitivity  = $auditData.SensitivityLabelId
```

```
    }
}

$results | Format-Table Timestamp, User, AppHost, GroundedFiles -AutoSize
$results | Export-Csv -Path ".\Copilot-Audit-Events.csv" -NoTypeInformation -Encoding utf8
```

---

5. Summary Checklist: Copilot Pre-Deployment Security

- [x] **Disable** `Everyone except external users` on sensitive SharePoint and OneDrive sites.
- [x] **Enable Restricted SharePoint Search (RSS)** if enterprise data hygiene is still underway.
- [x] **Deploy Purview Sensitivity Labels** with encryption to block unauthorized grounding.
- [x] **Turn on Purview AI Hub** in the Microsoft Purview portal to monitor prompt risk and sensitive info types (SITs).
- [x] **Schedule Weekly UAL Audits** using `Search-UnifiedAuditLog -RecordType CopilotInteraction`.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 03 OF 10

SharePoint Advanced Management (SAM): Data Access Governance (DAG), Restricted Access Control (RAC) & Stale Site Hygiene

Chapter 3 of AI for Microsoft 365 Administrators. Master SharePoint Advanced Management (SAM) unlocked by Copilot: Data Access Governance (DAG) reports, Restricted Access Control (RAC), Restricted Content Discovery (RCD), Site Access Reviews, and Inactive Site Policies with PowerShell.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch3-sharepoint-advanced-management-sam-dag-rac.html

🚨 CHAPTER 3 · LEARNING OBJECTIVES AI FOR M365 ADMINS · SHAREPOINT ADVANCED MANAGEMENT (SAM)

What You Will Master in Chapter 3

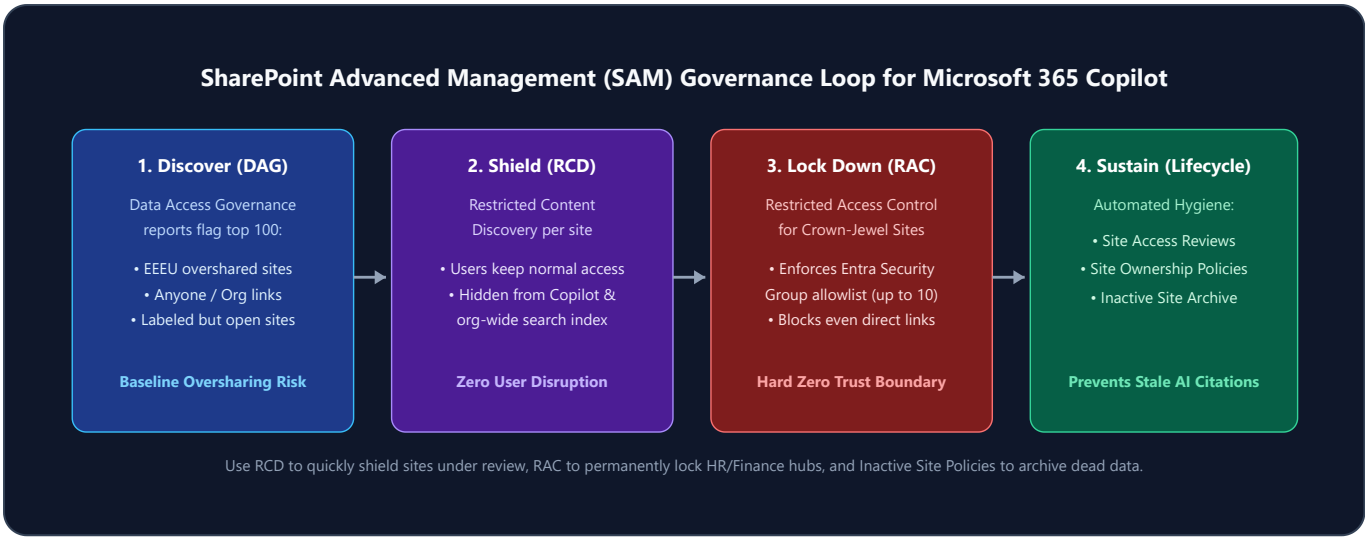
- **Copilot unlocks SAM:** Purchasing Microsoft 365 Copilot unlocks core **SharePoint Advanced Management (SAM)** governance controls directly inside the SharePoint Admin Center.
- **Data Access Governance (DAG) Reports:** Identify sites with `Everyone except external users` (EEEU), anonymous sharing links, and unlabeled sensitive libraries in minutes.
- **RAC vs. RCD (Know the Difference): Restricted Access Control (RAC)** hard-blocks site access to specific Entra security groups; **Restricted Content Discovery (RCD)** leaves normal permissions intact while excluding the site from Copilot and organization-wide search.
- **Stale Data Hygiene:** Configure **Inactive Site Policies** so Copilot stops citing obsolete 2019 policies and abandoned project sites.

Set-SPOSite -Identity "https://contoso.sharepoint.com/sites/Finance" -RestrictContentOrgWideSearch $true

In Chapter 2, we tackled immediate oversharing containment using **Restricted SharePoint Search (RSS)** (an allowlist of up to 100 approved sites). RSS is an effective emergency brake, but it is a temporary bridge—you cannot leave a 5,000-site enterprise locked to 100 sites forever.

To scale Microsoft 365 Copilot safely across your entire SharePoint Online estate, you need **SharePoint Advanced Management (SAM)**. Even better: Microsoft includes core SAM Copilot-governance capabilities at no extra charge for tenants with Microsoft 365 Copilot licenses.

In Chapter 3, we walk through the four SAM pillars—**DAG Reports**, **Restricted Content Discovery (RCD)**, **Restricted Access Control (RAC)**, and **Site Lifecycle / Access Reviews**—with SharePoint Online PowerShell commands ready for production.



---

1. Restricted Content Discovery (RCD) vs. Restricted Access Control (RAC)

Administrators frequently confuse **RCD** and **RAC**. Use this decision matrix before running PowerShell:

Control	What It Blocks	Does It Break User Bookmarks?	Best Use Case
Restricted Content Discovery (RCD)	Blocks the site from **Copilot grounding** and **organization-wide search** (unless the user directly opens or recently edited the file).	**No**	Quickly shielding messy or legacy sites while site owners complete access reviews.
Restricted Access Control (RAC)	Hard-blocks **all access** (browser, OneDrive sync, Teams files, AND Copilot) for anyone outside up to 10 designated Entra security groups.	**Yes** (for non-group members)	Crown-jewel sites: HR, Legal, M&A, Executive Board, Payroll.

---

2. Configuring RCD & RAC with SharePoint Online PowerShell

Connect to the SharePoint Online Management Shell (`Microsoft.Online.SharePoint.PowerShell` v16.0.25000+):

POWERSHELL

```powershell
# Connect as SharePoint Administrator
Connect-SPOService -Url "https://contoso-admin.sharepoint.com"

# ──────────────────────────────────────────────────────────────
# 1. Enable Restricted Content Discovery (RCD) on a sensitive site
#    Excludes the site from Copilot & org-wide search without breaking links
# ──────────────────────────────────────────────────────────────
Set-SPOSite -Identity "https://contoso.sharepoint.com/sites/Legacy-HR-Archive" `
```

```
    -RestrictContentOrgWideSearch $true

# Audit all sites currently shielded by RCD
Get-SPOSite -Limit All |
    Where-Object { $_.RestrictContentOrgWideSearch -eq $true } |
    Select-Object Title, Url, RestrictContentOrgWideSearch |
    Format-Table -AutoSize

# ─────────────────────────────────────────────────────────────
# 2. Enable Restricted Access Control (RAC) on a Crown-Jewel Finance Site
#    Only members of the specified Entra Security Group GUID can open or ground on it
# ─────────────────────────────────────────────────────────────
$financeGroupGuid = "11111111-2222-3333-4444-555555555555"

Set-SPOSite -Identity "https://contoso.sharepoint.com/sites/Executive-Finance" `
    -RestrictedAccessControl $true `
    -RestrictedAccessControlGroups @($financeGroupGuid)

# Audit all RAC-protected sites across the tenant
Get-SPOSite -Limit All |
    Where-Object { $_.RestrictedAccessControl -eq $true } |
    Select-Object Title, Url, RestrictedAccessControl, RestrictedAccessControlGroups |
    Format-Table -AutoSize
```

---

3. Running Data Access Governance (DAG) Reports & Site Access Reviews

Inside **SharePoint Admin Center → Reports → Data access governance**:

1. Run the **"Sharing links"** report (identifies sites with the highest count of *Anyone* and *People in your organization* links created in the last 28 days).
2. Run the **"Content shared with Everyone except external users (EEEU)"** report.
3. Click **Initiate site access review** directly from the report row: SAM automatically emails the site owner with a guided checklist to remove overshared groups, revoke stale links, or confirm business justification—shifting cleanup from central IT to the business owners who actually know who belongs in the site.

---

4. Eliminating "Stale AI Hallucinations" with Inactive Site Policies

Oversharing is only half the SharePoint problem—**stale data** is the other half. If your tenant has a 2019 travel expense policy sitting on an abandoned SharePoint site alongside your 2026 travel policy, Copilot may blend both documents together and give employees wrong reimbursement limits.

In **SharePoint Admin Center → Policies → Site lifecycle management**:

1. Create an **Inactive site policy** targeting sites with **no user activity for 180–365 days**.
2. Set the automated action to **Notify site owners** (monthly for 3 months), then **Move to Read-Only** or **Archive via Microsoft 365 Archive** if uncertified.
3. Create a **Site ownership policy** requiring every active SharePoint site to maintain at least **2 active owners** so no site becomes an unmonitored orphan.

---

5. Chapter 3 Lab Checklist

- [ ] Run DAG reports in SharePoint Admin Center and export the top 25 EEEU-exposed sites.
- [ ] Apply **Restricted Content Discovery (** `-RestrictContentOrgWideSearch $true` **)** to any legacy/unreviewed site before assigning Copilot licenses.
- [ ] Apply **Restricted Access Control (RAC)** to HR, Legal, and Executive Finance sites bound to dedicated Entra ID security groups.
- [ ] Enable **Site Lifecycle Inactive Policies** (365-day threshold) to archive stale documents out of Copilot's Semantic Index.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 04 OF 10

Microsoft Purview for AI: Sensitivity Labels, Copilot DLP & Administrative Units

How to build Microsoft Purview DLP policies that stop Microsoft 365 Copilot from processing sensitive content, and how to delegate them to regional admins with Entra administrative units using New-DlpCompliancePolicy and -PolicyRBACScopes.

Online Runbook: www.lab365.in/2026/11/purview-dlp-microsoft-365-copilot-administrative-units-powershell.html

🤖 AI EXECUTIVE BRIEF & TL;DR PURVIEW DLP · MICROSOFT 365 COPILOT

What you need to know

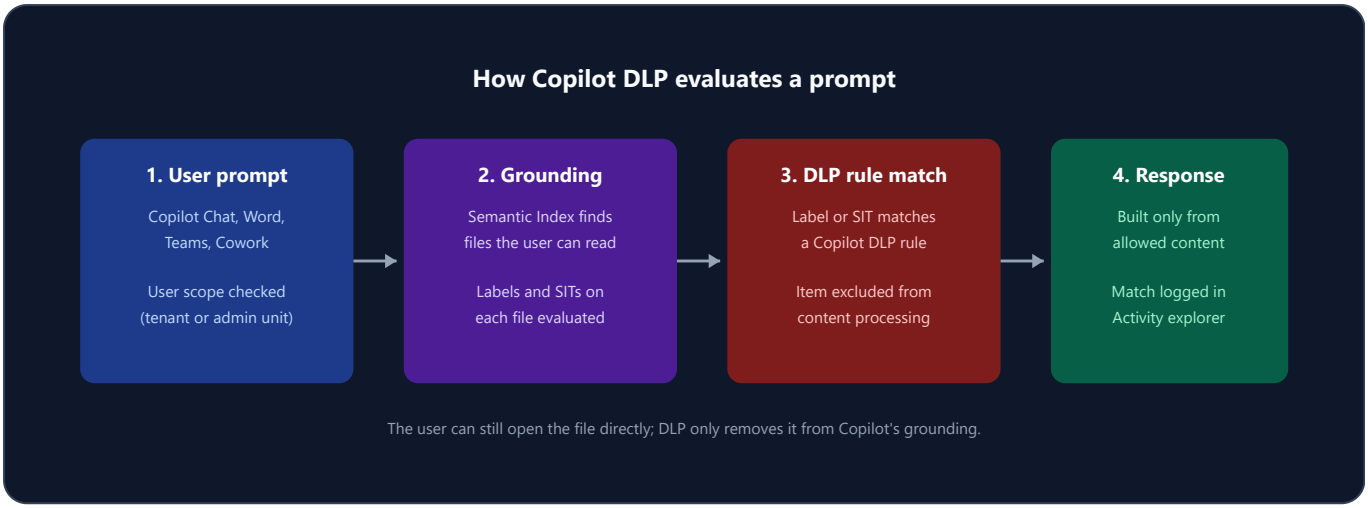
- **Copilot is its own DLP location.** Policies target it with the location ID `470f2276-e011-4e9d-a6ec-20768be3a4b0` and the `CopilotExperiences` enforcement plane.
- **The rule action is "exclude from processing".** Matching files and prompts aren't used to ground Copilot responses. Copilot DLP doesn't block the user from opening the file.
- **Regional teams can own their own policies.** `-PolicyRBACScopes` scopes a policy to an Entra administrative unit, so an EMEA admin only sees and manages EMEA policies.

Connect-IPPSSession; Get-DlpCompliancePolicy | Select Name, Mode, Workload

Most Copilot data-exposure problems are permission problems: Copilot can only ground on content the user can already open. Fixing oversharing is the long-term answer (see our Copilot tenant security runbook). But oversharing cleanup takes months. **Purview DLP for Microsoft 365 Copilot** is the control you can turn on this week. It stops Copilot from processing content that carries specific sensitivity labels or sensitive information types, even when the user technically has access.

This guide covers how the Copilot DLP location works, how to create the policy and rules in PowerShell, and how to delegate policies to regional compliance teams with **Entra administrative units**.

NOTE:
Lab status: The Graph and Autodiscover checks in today's posts were run against the LAB 365 dev tenant. The Purview cmdlets in this post were **not** lab-run: the lab host doesn't have the `ExchangeOnlineManagement` module installed yet. They follow Microsoft's documented syntax. Run them in a test policy (`-Mode TestWithoutNotifications`) first.



---

1. Prerequisites

Requirement	Detail
Licensing	Microsoft 365 Copilot for the users in scope, plus a Purview licence that includes DLP for Copilot (E5 or E5 Compliance).
Role	*Compliance Administrator* or *Compliance Data Administrator*. For admin-unit scoped policies, the regional admin needs the role **assigned scoped to that admin unit**.
Module	`ExchangeOnlineManagement` v3.x for `Connect-IPPSSession`.
Labels	Sensitivity labels already published, if you want label-based rules (recommended).

POWERSHELL

```powershell
Install-Module ExchangeOnlineManagement -Scope CurrentUser
Connect-IPPSSession -UserPrincipalName admin@contoso.com
```

---

2. Create a tenant-wide Copilot DLP policy

Copilot uses the `Applications` workload with a fixed location ID. The policy also needs the `CopilotExperiences` enforcement plane.

POWERSHELL

```powershell
# Microsoft 365 Copilot DLP location
$copilotLocation = @'
[{"Workload":"Applications","Location":"470f2276-e011-4e9d-a6ec-20768be3a4b0","Inclusions":
[{"Type":"Tenant","Identity":"All"}]}]
'@
```

```powershell
New-DlpCompliancePolicy -Name "Copilot - Block Highly Confidential" `
    -Comment "Exclude Highly Confidential content from Copilot grounding" `
    -Mode TestWithoutNotifications `
    -Locations $copilotLocation `
    -EnforcementPlanes @("CopilotExperiences")
```

Start in `TestWithoutNotifications` mode. That lets you see matches in Activity explorer without changing what users get back from Copilot.

---

3. Add the rule: exclude labelled content from processing

The rule matches a sensitivity label and tells Copilot not to process the matching item.

POWERSHELL

```powershell
# Find the label GUID
Get-Label | Select-Object DisplayName, ImmutableId

$labelId = "<ImmutableId of 'Highly Confidential'>"

New-DlpComplianceRule -Name "Copilot - Highly Confidential label" `
    -Policy "Copilot - Block Highly Confidential" `
    -ContentContainsSensitiveInformation @{
        operator = "And"
        groups   = @(@{
            operator = "Or"
            name     = "Default"
            labels   = @(@{ name = $labelId; type = "Sensitivity" })
        })
    } `
    -RestrictAccess @(@{ setting = "ExcludeContentProcessing"; value = "Block" })
```

To match on sensitive information types instead (for example credit card numbers in prompts), swap the label group for a `sensitivetypes` entry. If you have the choice, label-based rules are more predictable than pattern matching.

---

4. Delegate to regional teams with administrative units

Large tenants rarely want one central team owning every Copilot policy. Purview supports **administrative-unit scoped policies**:

1. Create the admin unit in Entra (e.g. `AU-EMEA`) and add its users. Dynamic membership on `country` works well.
2. Assign the EMEA compliance admin the *Compliance Administrator* role **scoped to** `AU-EMEA` in the Purview portal (*Roles & scopes → Role groups*).
3. Create the policy with `-PolicyRBACScopes` so only users in that unit are evaluated, and only scoped admins can see it.

POWERSHELL

```powershell
# Object ID of the Entra administrative unit
$auEmea = (Get-MgDirectoryAdministrativeUnit -Filter "displayName eq 'AU-EMEA'").Id

$emeaLocation = @'
[{"Workload":"Applications","Location":"470f2276-e011-4e9d-a6ec-20768be3a4b0","Inclusions":
[{"Type":"Tenant","Identity":"All"}]}]
'@

New-DlpCompliancePolicy -Name "Copilot - EMEA Customer Data" `
    -Mode TestWithoutNotifications `
    -Locations $emeaLocation `
    -EnforcementPlanes @("CopilotExperiences") `
    -PolicyRBACScopes $auEmea
```

> **IMPORTANT:**
> An admin-unit scoped policy only applies to users in that unit. Keep one **tenant-wide baseline** policy (section 2) so users outside every regional unit are still covered.

---

5. Audit what you have

POWERSHELL

```powershell
Get-DlpCompliancePolicy |
    Select-Object Name, Mode, Enabled, PolicyRBACScopes, WhenChanged |
    Format-Table -AutoSize

Get-DlpComplianceRule |
    Select-Object Name, ParentPolicyName, Disabled |
    Format-Table -AutoSize
```

Once matches look right in **Purview** → **Data loss prevention** → **Activity explorer** (filter on *Activity: DLP rule matched*, *Location: Microsoft 365 Copilot*), switch the policy to enforce:

POWERSHELL

```powershell
Set-DlpCompliancePolicy -Identity "Copilot - Block Highly Confidential" -Mode Enable
```

---

6. Rollout checklist

- [ ] Sensitivity labels published and applied to your most sensitive libraries
- [ ] Tenant-wide Copilot DLP baseline created in test mode
- [ ] Regional admin units created, with scoped Compliance Administrator role assignments
- [ ] Regional policies created with `-PolicyRBACScopes`
- [ ] Two weeks of Activity explorer review before switching to `-Mode Enable`

- [ ] Oversharing remediation still running in parallel. DLP buys time but doesn't replace fixing permissions

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 05 OF 10

Purview DSPM for AI & Shadow AI: Detecting Unsanctioned AI Tools, Endpoint DLP & Intune Edge Controls

Chapter 5 of AI for Microsoft 365 Administrators. Discover and govern Shadow AI (ChatGPT, Claude, Gemini, local coding agents like Copilot CLI and Claude Code) using Microsoft Purview DSPM for AI, Endpoint DLP, Defender local agent discovery, and Intune Microsoft Edge policies.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch5-purview-dspm-shadow-ai-local-agents-intune.html

🚨 CHAPTER 5 · LEARNING OBJECTIVES AI FOR M365 ADMINS · PURVIEW DSPM FOR AI & SHADOW AI

What You Will Master in Chapter 5

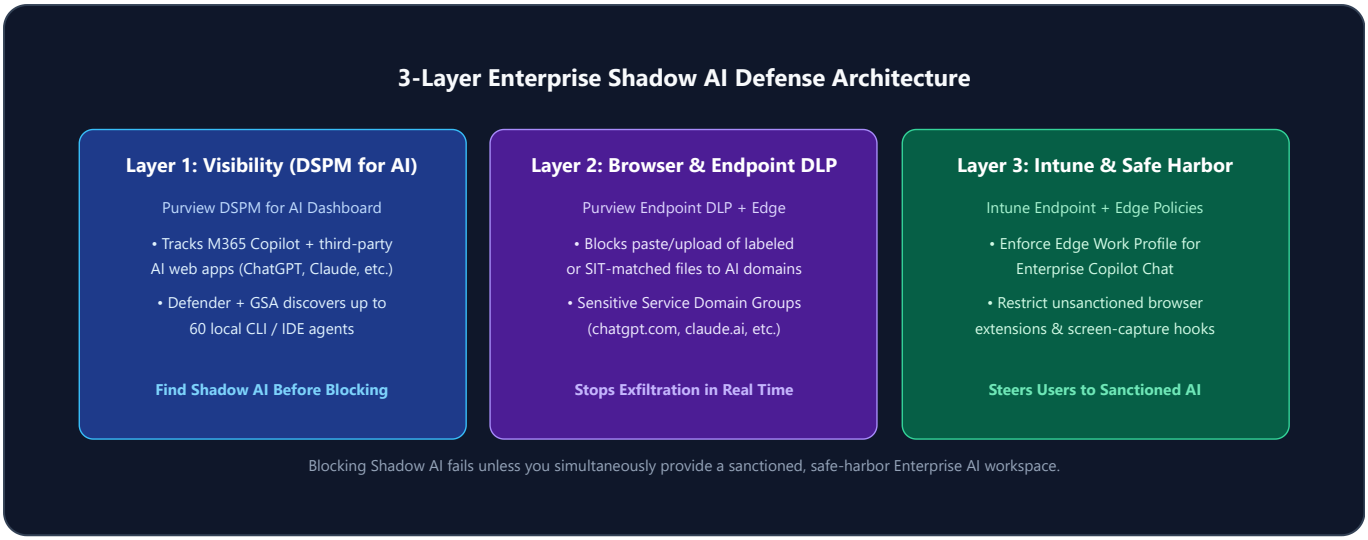
- **Shadow AI is bigger than Copilot:** Even if you haven't rolled out Microsoft 365 Copilot, employees are already pasting source code and spreadsheets into consumer ChatGPT, Gemini, Claude, and local coding agents.
- **Purview DSPM for AI:** Use Microsoft Purview's unified posture dashboard to track sensitive info types (SITs) entering both Microsoft Copilot and third-party AI web apps.
- **Local Agent Discovery (Ignite 2026):** Pair Microsoft Defender for Endpoint + Entra Global Secure Access to discover up to 60 local developer agents (such as GitHub Copilot CLI and Claude Code) running on corporate workstations.
- **Intune + Edge Guardrails:** Enforce Endpoint DLP browser restrictions and Intune Edge policies to block unauthenticated consumer AI uploads while steering users to sanctioned Enterprise Copilot Chat.

Connect-IPPSSession; Get-DlpCompliancePolicy | Where-Object { $_.EndpointDlpLocation -ne $null } | Select Name, Mode

In Chapters 1 through 4, we focused on governing **Microsoft 365 Copilot** inside your tenant boundary. But ask any CISO what keeps them awake at night in 2026, and the answer is **Shadow AI**:

- A finance analyst pasting an unreleased Q4 earnings table into a personal ChatGPT or Claude browser tab.
- A developer running an unsanctioned local coding agent (or an OS-level screen-capturing agent) on a corporate laptop with full read access to local Git repos and browser sessions.

You cannot solve Shadow AI with an Acceptable Use Policy PDF. You need **visibility** (Microsoft Purview **DSPM for AI** + Defender local agent discovery) and **technical enforcement** (**Endpoint DLP** + **Intune Edge policies**).



---

1. Enabling Purview DSPM for AI One-Click Policies

In the **Microsoft Purview portal → Solutions → DSPM for AI**:

1. **Verify Prerequisites:** Ensure Microsoft Purview Audit is enabled and Windows/macOS endpoints are onboarded to **Microsoft Defender for Endpoint / Purview Device Onboarding**.
2. **Activate One-Click Policies:** Enable *Detect sensitive info added to AI sites* (monitors clipboard paste and file uploads containing PII/PCI/source code to top generative AI web domains in Microsoft Edge and Chrome via the Purview extension) and *Insider Risk Management — Risky AI usage* (correlates high-volume prompt activity with resignation dates or mass SharePoint downloads).
3. **Review Activity Explorer:** Filter by `Activity: Uploaded to cloud / Pasted to browser` to see which departments are using unsanctioned AI tools—and why.

---

2. Blocking Sensitive Uploads to Unsanctioned AI Sites with Endpoint DLP

Rather than blanket-blocking `chatgpt.com` or `claude.ai` (which drives employees onto personal phones), **Purview Endpoint DLP** lets users visit AI sites for generic research while **blocking them from pasting or uploading sensitive corporate files**:

1. In **Purview → Settings → Data loss prevention → Endpoint DLP settings → Service domains**, create a **Sensitive Service Domain Group** named `Unsanctioned-Consumer-AI-Sites` containing `chatgpt.com`, `chat.openai.com`, `claude.ai`, `gemini.google.com`, `perplexity.ai`, and `deepseek.com`.
2. Target your **Endpoint DLP Policy** to **Devices** (`EndpointDlpLocation`) with rules matching **Confidential / Highly Confidential** sensitivity labels or corporate Sensitive Information Types (SITs).
3. Under **Actions → Upload to a restricted cloud service domain or access from an unallowed browser**, set `Unsanctioned-Consumer-AI-Sites` to **Block** (or **Block with override** during a 2-week user education phase).

Auditing Endpoint DLP Policies via PowerShell

```powershell
Connect-IPPSSession -UserPrincipalName admin@contoso.com

# Inspect all active DLP policies protecting Endpoint Devices
Get-DlpCompliancePolicy |
    Where-Object { $_.EndpointDlpLocation -ne $null } |
    Select-Object Name, Mode, Enabled, Priority, WhenChanged |
    Format-Table -AutoSize
```

---

3. Discovering Local Coding Agents & Governing Edge via Intune

With the rise of local autonomous agents (GitHub Copilot CLI, Claude Code, Codex, and OS-level screen-reading assistants), browser-only DLP is no longer enough:

1. **Defender + Entra Global Secure Access Local Agent Discovery:** Enable endpoint discovery in Microsoft Defender XDR and Entra Agent ID to inventory up to 60 local agent binaries running across Windows and macOS workstations—and require **Microsoft Execution Containers (MXC)** policies (covered in Chapter 7) for approved developer agents.
2. **Intune Settings Catalog Policies for Microsoft Edge:** Deploy a dedicated **Intune Configuration Profile (Windows 10/11 → Settings Catalog → Microsoft Edge)** with two guardrails:

* **Configure Copilot in Edge:** Enable **Microsoft 365 Copilot Chat** with **Enterprise Data Protection** (signed into Entra ID work profile) and disable consumer Bing/personal sidebar sign-ins. * **Extension Allowlist:** Configure `ExtensionInstallBlocklist = *` paired with `ExtensionInstallAllowlist` to block unverified third-party "AI summarizer" Chrome/Edge extensions that silently read DOM contents of Outlook Web and SharePoint pages.

---

4. Chapter 5 Lab Checklist

- [ ] Open **Purview → DSPM for AI** and verify endpoint telemetry is flowing from managed Windows/macOS devices.
- [ ] Create the `Unsanctioned-Consumer-AI-Sites` service domain group in Purview Endpoint DLP settings.
- [ ] Deploy an Endpoint DLP rule in **Audit / Test** mode for 14 days to measure real Shadow AI usage before switching to **Block**.
- [ ] Configure **Intune Settings Catalog** for Microsoft Edge to enforce work-profile Enterprise Copilot Chat and block unapproved AI browser extensions.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 06 OF 10

Governing Autonomous AI Agents: Entra Agent ID, Blueprints & Agent 365

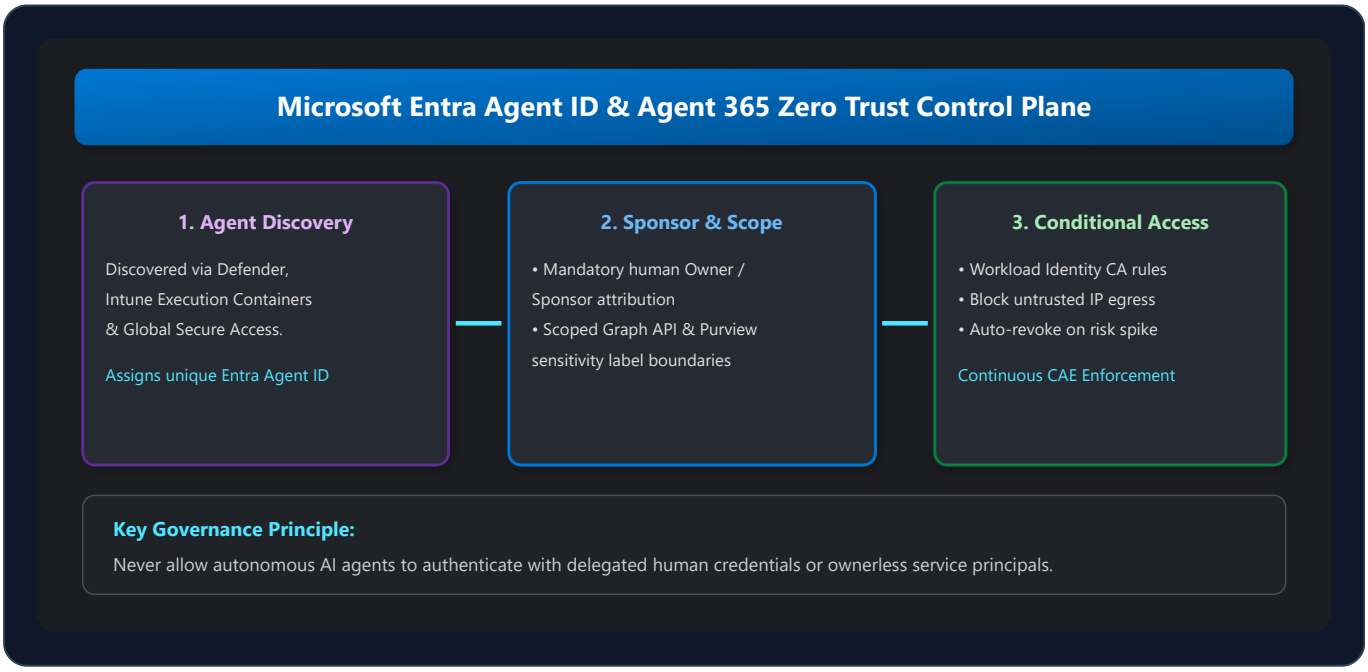
How to inventory, govern, and apply Conditional Access to autonomous AI agents using Microsoft Entra Agent ID (Ignite 2026 Preview), Agent 365, and Microsoft Graph PowerShell.

Online Runbook: www.lab365.in/2026/11/entra-agent-id-agent-365-audit-conditional-access-graph.html

At **Microsoft Ignite 2026**, Microsoft unveiled **Microsoft Entra Agent ID** (Preview) under the **Agent 365** control plane—marking a fundamental architectural shift in enterprise identity management. As organizations deploy autonomous AI agents across Microsoft 365 Copilot Studio, Azure AI Foundry, GitHub Copilot CLI, and endpoint execution containers, non-human agent identities now outnumber human employees in many tenants.

Without dedicated identity governance, AI agents often run under shared service principals or over-privileged user tokens, bypassing Multi-Factor Authentication (MFA) and creating massive lateral-movement blast radiuses. **Entra Agent ID** solves this by assigning every discovered cloud and local AI agent a distinct, managed workload identity with an explicit **human sponsor/owner**, separate **Conditional Access policies**, and continuous **Global Secure Access** telemetry.

Microsoft Entra Agent ID Governance Architecture



Step 1: Audit All Non-Human Workload & Agent Identities via Microsoft

Graph

Use Microsoft Graph PowerShell (`Application.Read.All` , `AuditLog.Read.All`) to inventory all service principals and workload/agent identities in your tenant—flagging any agent identity that lacks an assigned human owner or holds high-privilege application permissions:

POWERSHELL

```powershell
#Requires -Modules Microsoft.Graph.Authentication
<#
.SYNOPSIS
    Audits Entra ID Service Principals & AI Agent Identities for missing owners and high-risk app roles.
#>

Connect-MgGraph -Scopes "Application.Read.All", "Directory.Read.All"

Write-Host "[*] Querying first-party and custom Enterprise Application / Agent Principals ... " -
ForegroundColor Cyan
$Uri = "https://graph.microsoft.com/v1.0/servicePrincipals?`$filter=accountEnabled eq
true&`$select=id,appId,displayName,servicePrincipalType,tags,createdDateTime&`$top=500"

$Principals = [System.Collections.Generic.List[object]]::new()
do {
    $Page = Invoke-MgGraphRequest -Method GET -Uri $Uri
    if ($Page.value) { $Principals.AddRange($Page.value) }
    $Uri = $Page.'@odata.nextLink'
} while ($Uri)

$Audit = [System.Collections.Generic.List[PSCustomObject]]::new()

foreach ($Sp in $Principals) {
    # Skip built-in Microsoft first-party resource APIs unless tagged as Agent/Custom
    if ($Sp.tags -notcontains "WindowsAzureActiveDirectoryIntegratedApp" -and $Sp.servicePrincipalType -eq
"Legacy") { continue }

    $Owners = (Invoke-MgGraphRequest -Method GET -Uri
"https://graph.microsoft.com/v1.0/servicePrincipals/$($Sp.id)/owners?`$select=userPrincipalName").value
    $AppRoles = (Invoke-MgGraphRequest -Method GET -Uri
"https://graph.microsoft.com/v1.0/servicePrincipals/$($Sp.id)/appRoleAssignments").value

    $Audit.Add([PSCustomObject]@{
        DisplayName          = $Sp.displayName
        AppId                = $Sp.appId
        ObjectId             = $Sp.id
        PrincipalType        = $Sp.servicePrincipalType
        OwnerCount           = $Owners.Count
        Owners               = ($Owners.userPrincipalName -join "; ")
        AppPermissionCount   = $AppRoles.Count
        GovernanceStatus     = if ($Owners.Count -eq 0) { "CRITICAL: Ownerless Identity" } else {
"Governed" }
    })
}

$Audit | Sort-Object OwnerCount, AppPermissionCount -Descending |
    Export-Csv ".\Entra-Agent-Workload-Identity-Audit.csv" -NoTypeInformation -Encoding UTF8

Write-Host "[+] Exported $($Audit.Count) workload/agent identities to Entra-Agent-Workload-Identity-
Audit.csv" -ForegroundColor Green
```

Step 2: Enforce Conditional Access for Workload & Agent Identities

With **Microsoft Entra Conditional Access for Workload Identities**, create a dedicated policy scoping non-human principals:

1. In the **Microsoft Entra admin center** (`entra.microsoft.com`), navigate to **Protection** > **Conditional Access** > **Policies** > **New policy**.
2. Under **Workload identities**, select **All owned service principals and Agent IDs** (or target a security/custom attribute set for AI Agents).
3. Under **Conditions** > **Locations** or **Workload identity risk**:

- **Block access** when **Workload identity risk** is `Medium` or `High`. - **Block access** from any IP outside your approved **Named Locations** (Azure VNet NAT IPs / Global Secure Access egress).

For deeper zero-trust sandboxing of dynamic AI agent permissions and cost guardrails, read our companion Auditing Copilot Credit Spending Policies & Entra Agent ID Inactivity Runbook, Governing AI Agent Dynamic Permissions in Entra ID Runbook, or grab the complete 10-chapter **AI for Microsoft 365 Administrators Book Edition (Web + PDF)**.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 07 OF 10

Sandboxing Local & Cloud AI Agents: Microsoft Execution Containers (MXC 1.0) & Dynamic Permissions

What Microsoft Execution Containers (MXC) 1.0 does for AI agent security, how it fits with Entra Agent ID blueprints, and how to inventory agent identity blueprints with Microsoft Graph PowerShell.

Online Runbook: www.lab365.in/2026/11/sandboxing-ai-agents-microsoft-execution-containers-entra-agent-identity-blueprints.html

🤖 **AI EXECUTIVE BRIEF & TL;DR MXC 1.0 · ENTRA AGENT ID**

What you need to know

- **An agent can't police itself.** MXC puts an OS-enforced boundary (files, network, UI) around agents and tools, defined in a declarative JSON policy.
- **Containment and identity are two separate layers.** MXC limits what an agent can touch *on the device*. Entra Agent ID (blueprints and agent identities) limits what it can reach *in the cloud*. You need both.
- **Start with an inventory.** List every agent identity blueprint in your tenant with Graph before you write any policy.

Invoke-MgGraphRequest GET "v1.0/applications/microsoft.graph.agentIdentityBlueprint"

Coding agents, browser agents and "do it for me" assistants now run shell commands, edit files and call APIs on users' machines. The security question has shifted from *what can the model say* to **what can the agent do**. Microsoft's answer has two parts:

- **Microsoft Execution Containers (MXC)**, now at a stable 1.0 release: an open-source, policy-driven sandbox for running agents and their tools on Windows, Linux and macOS.
- **Microsoft Entra Agent ID**: first-class identities for agents, created from **agent identity blueprints**, so cloud access is governed like any other identity.

This post explains how the two fit together and what an infrastructure team should do first. For the identity-governance side in depth, see our Entra Agent ID & Agent 365 audit guide and AI agent dynamic permissions runbook.



---

1. What MXC actually does

MXC separates an agent's execution environment from the host using a **declarative policy**. The operating system enforces that policy at runtime, not the agent. If a coding agent tries to write outside its allowed directory or call a network endpoint that isn't on its list, the OS denies it ("Access is denied" or a dropped connection), whatever the model decided.

Key points from the 1.0 release:

- **One abstraction, several backends.** MXC picks the isolation primitive that fits the task: lightweight process containers (AppContainer on Windows, Seatbelt on macOS, Bubblewrap on Linux), session containers that hide the user's desktop, clipboard and input, WSL or microVM isolation for untrusted code, and cloud instances through Windows 365 for Agents.
- **SDKs** for Rust, .NET and TypeScript/Node.js, with versioned entry points such as `@microsoft/mxc-sdk/v1`, so agent harnesses can request containment directly.
- **Ecosystem adoption.** GitHub Copilot CLI, OpenAI Codex, NVIDIA OpenShell, Meta Muse, Claude Code and others have announced support or plans to support it.

What a policy expresses

The exact schema is in the MXC repository. Conceptually, a policy for a coding agent looks like this:

JSON

```
{
  "_comment": "Illustrative only - check the MXC repo for the exact v1 schema",
  "filesystem": {
    "read":  ["C:\\src\\project"],
    "write": ["C:\\src\\project\\out"]
  },
  "network": {
    "egress": ["api.github.com:443", "registry.npmjs.org:443"]
  },
```

```
  "ui": { "clipboard": false, "desktop": false }
}
```

> **NOTE:**
> The snippet above is an illustration of the *kinds* of rules MXC enforces, not the official schema. Use the published v1 schema from the MXC project when you write real policies.

---

2. Where Entra Agent ID fits

MXC stops an agent from wandering around the device. It doesn't decide which mailboxes, sites or APIs the agent can call in Microsoft 365. That is Entra's job.

An **agent identity blueprint** (`microsoft.graph.agentIdentityBlueprint`) is a specialised application object. It holds the shared configuration (display name, publisher, app roles, credentials, token settings) for a family of agents. Individual **agent identities** are created from it, and a blueprint principal in your tenant issues their tokens. Every blueprint must have **sponsors**: named humans who are accountable for it.

The relevant Graph permissions and roles:

Task	Graph permission	Entra role
Read blueprints	`AgentIdentityBlueprint.Read.All`	Agent ID Administrator / Global Reader
Create blueprints	`AgentIdentityBlueprint.Create`	Agent ID Developer or Administrator
Manage blueprints	`AgentIdentityBlueprint.ReadWrite.All`	Agent ID Administrator

---

3. Inventory agent identity blueprints with PowerShell

POWERSHELL

```powershell
Connect-MgGraph -Scopes "AgentIdentityBlueprint.Read.All"

$uri = "https://graph.microsoft.com/v1.0/applications/microsoft.graph.agentIdentityBlueprint" +
       "?`$select=id,appId,displayName,createdDateTime"

$blueprints = @()
do {
    $page = Invoke-MgGraphRequest -Method GET -Uri $uri
    $blueprints += $page.value
    $uri = $page.'@odata.nextLink'
} while ($uri)

$blueprints |
    Select-Object displayName, appId, createdDateTime |
    Sort-Object createdDateTime -Descending |
    Format-Table -AutoSize
```

```
"Total agent identity blueprints: $($blueprints.Count)"
```

Lab result (LAB 365 dev tenant): The same call with our existing automation app returned **HTTP 403**. The app has `Directory.ReadWrite.All` but not `AgentIdentityBlueprint.Read.All`. Broad directory permissions **don't** grant access to agent blueprints. Inventory scripts need the dedicated permission.

Check each blueprint has a sponsor

A blueprint without a sponsor is an orphaned agent waiting to happen:

POWERSHELL

```powershell
foreach ($bp in $blueprints) {
    $sponsors = Invoke-MgGraphRequest -Method GET `
        -Uri
"https://graph.microsoft.com/v1.0/applications/$($bp.id)/microsoft.graph.agentIdentityBlueprint/sponsors"
    if (-not $sponsors.value) {
        Write-Warning "No sponsor: $($bp.displayName) ($($bp.appId))"
    }
}
```

---

4. A practical starting plan

1. **Inventory.** List blueprints and agent identities (section 3) and the agent tools developers already run locally (Copilot CLI, Codex, Claude Code and similar).
2. **Assign accountability.** Every blueprint gets at least one sponsor. Unsponsored blueprints go on a disable list.
3. **Contain on devices.** For developer agents that support MXC, require a containment policy that limits writes to the project folder and egress to known package and API hosts.
4. **Restrict in the cloud.** Grant agent identities the narrowest app roles possible and target them with Conditional Access for agents.
5. **Monitor.** Watch Entra sign-in and audit logs for agent identities, and Defender / Sentinel for agent processes breaking their containment policy.

---

5. Summary

Layer	Control	Owner
Device	MXC containment policy	Endpoint / platform team
Identity	Agent identity blueprints, sponsors, app roles	Identity team
Access	Conditional Access for agents	Identity / security team
Visibility	Agent 365, Entra logs, Defender, Sentinel	SOC

> **Continue Your AI Governance Journey:** Audit dormant agent blueprints and lock down PAYG credit caps in our Copilot Credit Spending Policies & Entra Agent ID Inactivity Runbook, or download the complete 10-chapter **AI for Microsoft 365 Administrators Playbook (Web + PDF)**.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 08 OF 10

Copilot Studio, SharePoint Agents & Connector Write Actions: Power Platform & API Governance

Chapter 8 of AI for Microsoft 365 Administrators. Govern custom Copilot Studio agents, SharePoint agents, and GA Connector Write Actions using Power Platform Managed Environments, DLP connector groups, environment routing, and Integrated Apps approval workflows.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch8-copilot-studio-sharepoint-agents-power-platform-governance.html

🚨 CHAPTER 8 · LEARNING OBJECTIVES AI FOR M365 ADMINS · COPILOT STUDIO & POWER PLATFORM

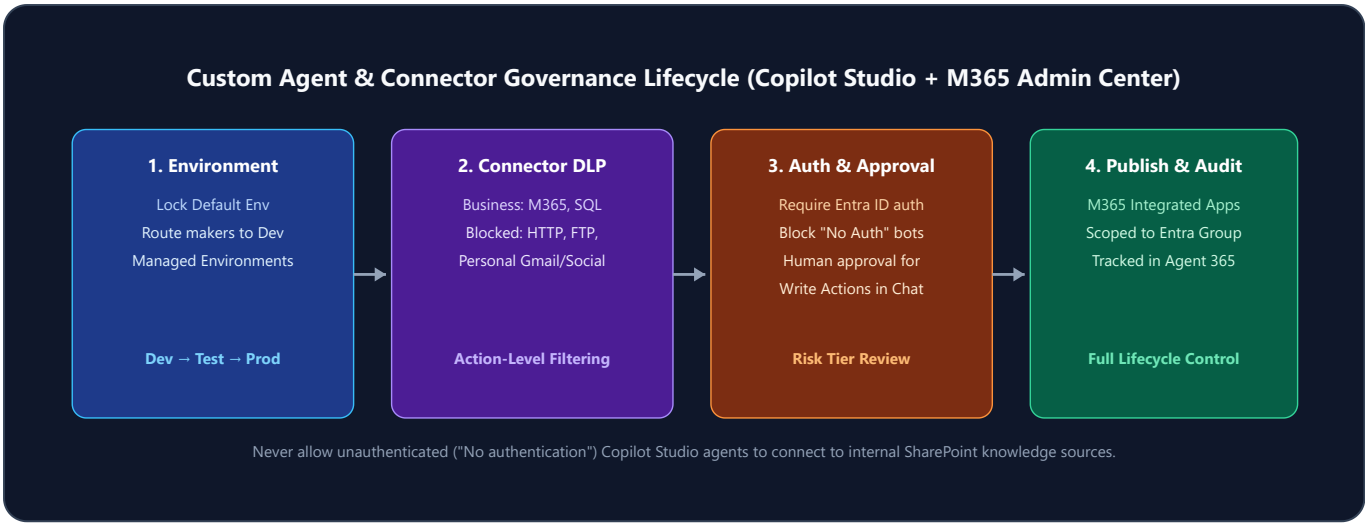
What You Will Master in Chapter 8

- **The Write-Action Risk Shift:** With **Connector Write Actions** GA in Copilot Chat and Cowork, custom agents no longer just *read* data—they can create ServiceNow tickets, update Salesforce records, send emails, and modify SharePoint lists.
- **Environment Strategy:** Lock down the **Default Environment**, enable **Environment Routing** for makers, and enforce Dev → Test → Prod Managed Environments for custom Copilot Studio agents.
- **Power Platform DLP Policies:** Classify connectors into *Business*, *Non-Business*, and *Blocked* buckets so a custom agent cannot read internal SharePoint data and POST it to an unapproved external webhook.
- **Integrated Apps Control Plane:** Restrict who can publish, share, and install declarative and custom agents inside the Microsoft 365 Admin Center.

Get-AdminPowerAppEnvironment | Select DisplayName, EnvironmentType, Location

Out-of-the-box Microsoft 365 Copilot is read-heavy. **Copilot Studio** and **SharePoint Agents** change the risk profile completely: any business user with a maker license can build a custom agent grounded on internal SharePoint libraries and wire it to **Power Platform connectors**—including live **Write Actions** that mutate external SaaS systems or internal databases.

Without Power Platform and M365 Integrated Apps guardrails, "citizen-built" agents quickly become unmanaged shadow automation with broad data access. Chapter 8 gives you the complete governance architecture.



---

1. Auditing Power Platform Environments & DLP Policies with PowerShell

Use the `Microsoft.PowerApps.Administration.PowerShell` module to inventory all environments where makers can create Copilot Studio agents and verify DLP policy coverage:

POWERSHELL

```powershell
# Install-Module -Name Microsoft.PowerApps.Administration.PowerShell -Scope CurrentUser
Add-PowerAppsAccount

Write-Host "=== 1. Power Platform Environments ===" -ForegroundColor Cyan
Get-AdminPowerAppEnvironment |
    Select-Object DisplayName, EnvironmentName, EnvironmentType, Location,
        @{N='IsDefault';E={$_.Internal.properties.isDefault}} |
    Format-Table -AutoSize

Write-Host "=== 2. Power Platform DLP Policies (Connector Boundaries) ===" -ForegroundColor Cyan
Get-AdminDlpPolicy |
    Select-Object DisplayName, PolicyName, DefaultConnectorClassification,
        @{N='EnvType';E={$_.EnvironmentType}} |
    Format-Table -AutoSize
```

---

2. Securing Connector Write Actions & Preventing Data Exfiltration

With **Connector Write Actions** GA in Copilot Chat and Cowork, enforce three non-negotiable Power Platform DLP rules:

1. **Block Cross-Boundary Data Mixing:** Put Microsoft 365 connectors (`SharePoint` , `Office 365 Outlook` , `Teams` , `Dataverse` , `SAP` , `ServiceNow`) in the **Business** group. Put `HTTP` , `HTTP with Microsoft Entra ID` , `Custom Connectors` , `Gmail` , and `Dropbox` in **Blocked** inside the Default Environment.
2. **Use Connector Action Control (Granular Read vs. Write):** On high-risk connectors (like `SharePoint` , `SQL Server` , or `ServiceNow`), use Power Platform DLP **Connector Action Control** to explicitly allow `Get/List` read actions while blocking destructive `Delete` or `Create/Update` actions in non-production environments.

3. **Mandate Human-in-the-Loop Confirmation:** Require explicit user confirmation prompts before any agent executes a Write Action on a user's behalf.

---

3. Controlling SharePoint Agents & M365 Integrated Apps

In **Microsoft 365 Admin Center** → **Copilot** → **Agents & Connectors** (and **Settings** → **Integrated apps**):

- **Restrict Org-Wide Publishing:** Require **Admin Approval** before any custom Copilot Studio agent can be published to the tenant catalog.
- **Control SharePoint Agents (`.agent` files):** Remember that any user with *Edit* permission on a SharePoint site can click **Create a Copilot agent** on a document library. Because SharePoint Agents consume **Pay-As-You-Go (PAYG)** messages for unlicensed users (covered in Chapter 10), govern who can create them via SharePoint site permissions and Monitor usage in the Agent 365 dashboard.

---

4. Chapter 8 Lab Checklist

- [ ] Audit all Power Platform environments and verify a strict DLP policy covers the **Default Environment**.
- [ ] Block **"No authentication"** and **"Share without authentication"** in Copilot Studio tenant governance settings.
- [ ] Configure **Connector Action Control** to restrict destructive Write Actions in dev/default environments.
- [ ] Require **Admin Approval** for publishing custom agents in **M365 Admin Center** → **Integrated Apps**.

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 09 OF 10

Microsoft Security Copilot & Sentinel: AI-Assisted Threat Hunting & KQL Incident Triage

Chapter 9 of AI for Microsoft 365 Administrators. Use Microsoft Security Copilot, Microsoft Sentinel data lakes, and Defender XDR KQL queries to accelerate phishing triage, hunt for rogue local AI agents, and automate incident response with SCU cost controls.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch9-security-copilot-sentinel-kql-threat-hunting.html

🚨 CHAPTER 9 · LEARNING OBJECTIVES AI FOR M365 ADMINS · SECURITY COPILOT & DEFENDER XDR

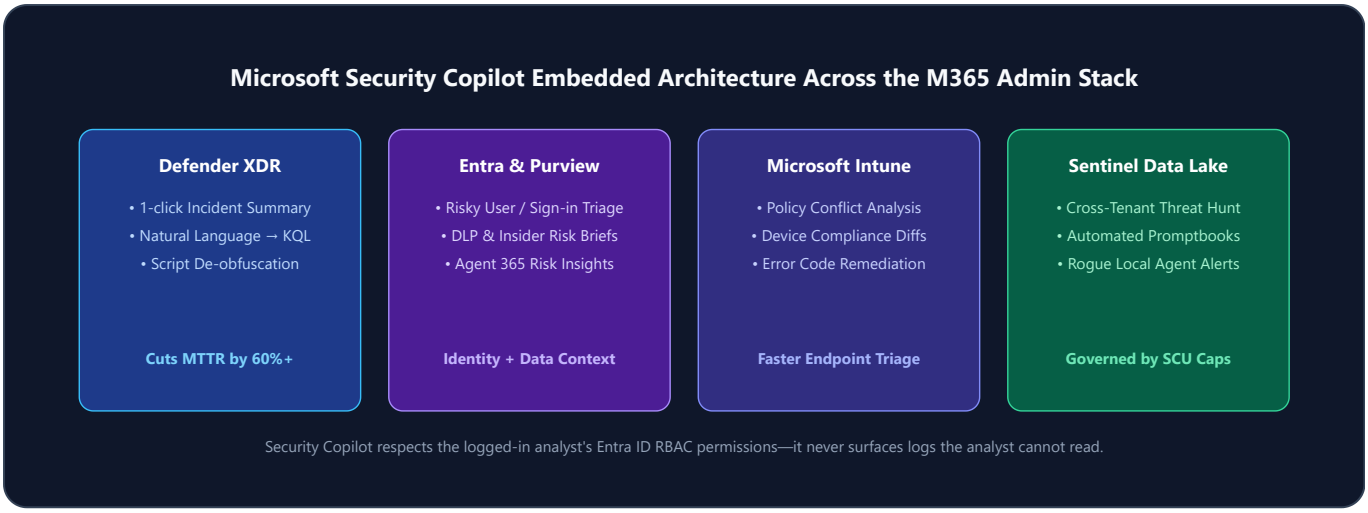
What You Will Master in Chapter 9

- **Embedded vs. Standalone Security Copilot:** Where M365 admins get the fastest ROI—embedded inside **Defender XDR**, **Microsoft Entra**, **Purview**, and **Intune** without switching portals.
- **SCU Capacity Planning:** How Security Compute Units (SCUs) are billed, how to prevent runaway promptbook consumption, and how to configure overage limits.
- **Ready-to-Run KQL Hunting Queries:** Hunt for anomalous Copilot grounding spikes, rogue local AI agent binaries, and post-outage admin phishing clicks across Defender + Sentinel.
- **Promptbooks for Tier-1/Tier-2 Triage:** Chain repeatable multi-step investigation prompts to turn a 45-minute BEC investigation into a 3-minute guided summary.

CloudAppEvents | where ActionType == "CopilotInteraction" | summarize Prompts=count() by AccountDisplayName

While Chapters 1–8 focused on **securing AI**, Chapter 9 flips the lens: **using AI to secure your Microsoft 365 tenant**.

Microsoft Security Copilot connects directly into **Microsoft Defender XDR**, **Microsoft Sentinel**, **Entra ID**, **Purview**, and **Intune**. Instead of manually writing complex 40-line KQL joins during a live Business Email Compromise (BEC) incident or hunting for a misconfigured local AI agent, administrators can summarize incidents, reverse-engineer obfuscated PowerShell payloads, and generate validated KQL hunting queries in seconds.



---

1. High-Signal KQL Queries for AI & Agentic Threat Hunting

Run these KQL queries in **Microsoft Defender XDR → Hunting → Advanced hunting** (or trigger them inside a Security Copilot Promptbook):

Query 1: Detect Anomalous Copilot Prompt Spikes (Potential Automated Scraping)

KQL

```
CloudAppEvents
| where Timestamp > ago(24h)
| where ActionType == "CopilotInteraction"
| summarize PromptCount = count(), DistinctApps = dcount(Application) by AccountDisplayName,
AccountObjectId, bin(Timestamp, 1h)
| where PromptCount > 60
| sort by PromptCount desc
```

Query 2: Hunt for Unsanctioned Local AI Coding Agents & Suspicious Subprocess Spawns

KQL

```
DeviceProcessEvents
| where Timestamp > ago(7d)
| where FileName in~ ("claude.exe", "codex.exe", "ollama.exe", "lmstudio.exe", "node.exe")
| where ProcessCommandLine has_any ("copilot", "claude", "openai", "anthropic", "mcp-server")
| summarize Executions = count(), Devices = dcount(DeviceName) by AccountName, FileName,
InitiatingProcessFileName
| sort by Executions desc
```

---

2. Building a Repeatable BEC & Phishing Triage Promptbook

Instead of typing ad-hoc prompts during an incident, save a **Security Copilot Promptbook** (`M365-Admin-BEC-Triage`) that runs 4 chained steps automatically for any suspicious UPN:

1. "*Summarize all Entra ID sign-in failures, impossible travel alerts, and newly registered MFA methods for* `<UPN>` *in the last 72 hours.*"
2. "*List any new Exchange Online inbox rules (* `New-InboxRule` *), forwarding settings, or OAuth app consents granted by* `<UPN>` *.*"
3. "*Check whether* `<UPN>` *triggered any Purview DLP or* `CopilotInteraction` *events touching Confidential-labeled files.*"
4. "*Generate a 5-bullet executive incident brief and recommended PowerShell containment commands (* `Revoke-MgUserSignInSession` *).*"

---

3. Managing Security Compute Units (SCUs) Without Bill Shock

Security Copilot is provisioned in hourly **Security Compute Units (SCUs)** billed to an Azure subscription:

- Start with **1–2 provisioned SCUs** for a 3–5 person SOC/M365 security team, and enable **overage SCUs with a strict monthly cap** so burst incident investigations don't stall while preventing runaway charges.
- Use **Role-Based Access Control (RBAC)** inside Security Copilot (*Settings → Role assignment*) so only designated SOC analysts and M365 Security Admins can invoke SCU-consuming plugins and promptbooks.

---

4. Chapter 9 Lab Checklist

- [ ] Verify Security Copilot RBAC assignments and cap overage SCUs in Azure Cost Management.
- [ ] Run the two KQL hunting queries above in Defender XDR Advanced Hunting to baseline Copilot volume and local agent processes.
- [ ] Create and test one repeatable **Promptbook** for BEC / phishing triage (see also our Post-Outage Defender KQL Hunting Runbook).

AI FOR MICROSOFT 365 ADMINISTRATORS · CH 10 OF 10

Copilot Cost Management (PAYG Billing & Caps), CopilotInteraction Auditing & ROI Analytics

Chapter 10 (Final Chapter) of AI for Microsoft 365 Administrators. Master Microsoft 365 Copilot seat vs. Pay-As-You-Go (PAYG) billing architecture, Azure budget caps, Viva Insights Copilot Analytics, and automated CopilotInteraction license reclamation with PowerShell.

Online Runbook: www.lab365.in/2026/11/ai-for-m365-admins-ch10-copilot-payg-billing-cost-caps-ual-auditing-roi.html

🚨 CHAPTER 10 · FINAL CHAPTER · LEARNING OBJECTIVES AI FOR M365 ADMINS · COST MANAGEMENT, PAYG & DAY-2 ROI

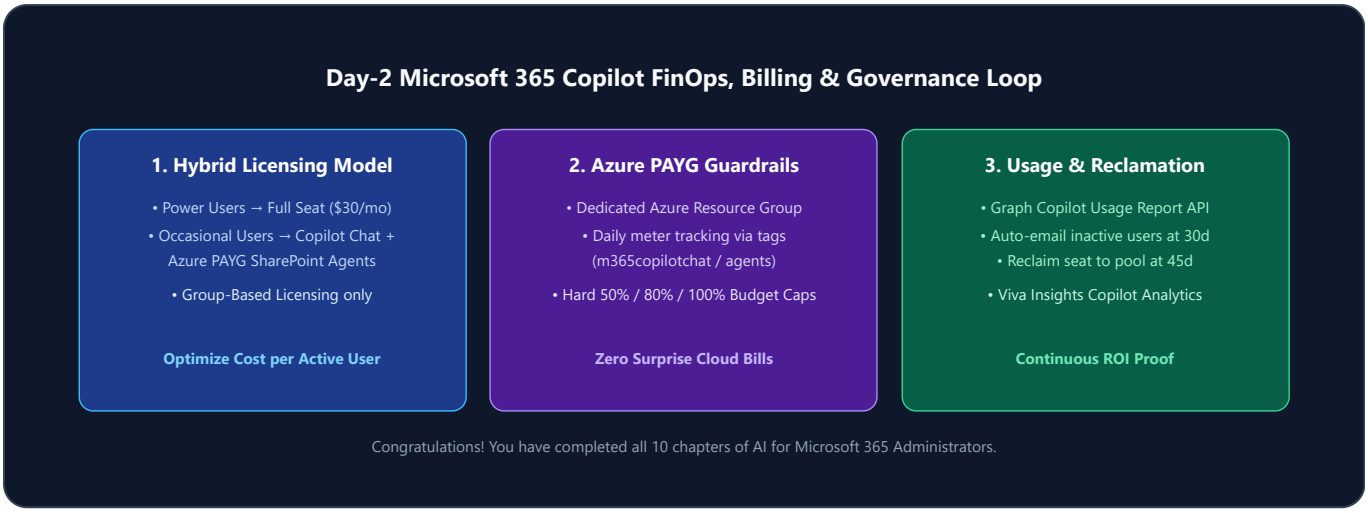
What You Will Master in Chapter 10

- **Seat vs. Pay-As-You-Go (PAYG) Strategy:** Assign paid full seats to heavy daily users (Sales, Legal, Engineering, Execs) and enable Azure-backed **PAYG billing** for occasional users accessing SharePoint Agents, Copilot Chat extensions, and Copilot Retrieval APIs.
- **Preventing Runaway PAYG Spend:** Link an Azure subscription + dedicated Resource Group in *M365 Admin Center → Copilot → Cost Management*, filter by tags like `m365copilotchat`, and enforce hard Azure Cost Management budget alerts and spending caps.
- **Automated License Reclamation:** Identify inactive Copilot license holders (0 active days in 45+ days) via Microsoft Graph usage reports and reclaim $30/month seats automatically.
- **Continuous Day-2 Operations:** Pair monthly `CopilotInteraction` audits with SAM access reviews so your security posture never drifts.

```
Invoke-MgGraphRequest GET "v1.0/reports/getMicrosoft365CopilotUsageUserDetail(period='D30')"
```

Deploying Microsoft 365 Copilot safely gets you to Day 1. Proving measurable business value—and stopping unused $30/month seats or uncapped **Pay-As-You-Go (PAYG)** agent meters from blowing up your IT budget—is what makes you a trusted enterprise AI architect on Day 2.

In this final chapter of *AI for Microsoft 365 Administrators*, we build your complete **FinOps, Auditing, and Adoption** operating model.



---

1. Configuring Pay-As-You-Go (PAYG) Billing & Budget Caps

When enabling **SharePoint Agents**, **Copilot Chat extensions**, or **Copilot Retrieval APIs** under consumption billing:

1. Create a dedicated Azure Resource Group (e.g., `rg-m365-copilot-payg-prod`) inside a subscription in the **same Entra ID tenant**.
2. In **Microsoft 365 Admin Center → Copilot → Billing & usage (Cost Management)**, link the Azure subscription and resource group, and scope the PAYG billing policy to an Entra security group (not `All Users` on Day 1!).
3. In the **Azure Portal → Cost Management + Billing → Budgets**, create a monthly budget scoped to `rg-m365-copilot-payg-prod` with automated email/webhook alerts at **50%**, **80%**, and **100%** forecasted spend.

---

2. Automated Inactive Copilot License Reclamation Script (Microsoft Graph)

Pull the official 30-day **Microsoft 365 Copilot Usage User Detail** report via Microsoft Graph (`Reports.Read.All`) to flag licensed users with zero recent activity so you can coach them—or reclaim and reassign their licenses:

POWERSHELL

```powershell
<#
.SYNOPSIS
    Audit-InactiveCopilotLicenses.ps1
    Identifies M365 Copilot users with no recorded usage in the past 30 days.
#>

Connect-MgGraph -Scopes "Reports.Read.All" -NoWelcome

$csvPath = ".\Copilot-Usage-30Day.csv"
Invoke-MgGraphRequest -Method GET `
    -Uri "https://graph.microsoft.com/v1.0/reports/getMicrosoft365CopilotUsageUserDetail(period='D30')" `
    -OutputFilePath $csvPath

$usage = Import-Csv -Path $csvPath
```

```powershell
$inactive = $usage | Where-Object { [string]::IsNullOrWhiteSpace($_.'Last Activity Date') }

Write-Host "Total Copilot Licensed Users Analyzed : $($usage.Count)" -ForegroundColor Cyan
Write-Host "Zero Activity in Last 30 Days         : $($inactive.Count)" -ForegroundColor Yellow

$inactive |
    Select-Object 'User Principal Name', 'Display Name', 'Last Activity Date' |
    Format-Table -AutoSize
```

---

3. Your Complete 10-Chapter Day-2 Governance Cadence

Cadence	Operational Task	Owner
Daily	Review Security Copilot / Defender XDR alerts for anomalous `CopilotInteraction` spikes and local agent detections	SOC / Security Admin
Weekly	Check Azure Cost Management PAYG meters (`m365copilotchat`) & Purview DSPM for AI Shadow AI trends	AI Admin / FinOps
Monthly	Run `Audit-InactiveCopilotLicenses.ps1` to reclaim idle seats; review Entra Agent ID sponsors & inactive blueprints	M365 / Identity Admin
Quarterly	Trigger SAM **Site Access Reviews** and verify **Restricted Access Control (RAC)** on HR/Legal/Finance hubs	SharePoint Admin

---

🎉 You Completed Book 1: *AI for Microsoft 365 Administrators*!

All 10 chapters are now live on **LAB 365 Learning Hub (`/learn/`)**. Bookmark the hub as we continue building **Book 2: Mastering Microsoft 365 & Endpoint Administration (with Intune)**!