packt

Unified SecOps Playbook

End-to-end enterprise security with Microsoft Sentinel, Defender XDR, and Security Copilot



JOSE LAZARO | MARCUS BURNAP

Foreword by Rod Trent,
Author of fiction and technical books, Microsoft Senior Product Manager for Security and AI

Unified SecOps Playbook

End-to-end enterprise security with Microsoft Sentinel, Defender XDR, and Security Copilot

Jose Lazaro

Marcus Burnap

‹packt›

Unified SecOps Playbook

Copyright © 2025 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the authors, nor Packt Publishing or its dealers and distributors, will be held liable for any damages caused or alleged to have been caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

Portfolio Director: Vijin Boricha

Relationship Lead: Prachi Sawant

Project Manager: Gandhali Raut

Content Engineer: Swathi Ajayakumar

Technical Editor: Nithik Cheruvakodan

Copy Editor: Safis Editing

Indexer: Pratik Shirodkar

Proofreader: Swathi Ajayakumar

Production Designer: Aparna Bhagat

Growth Lead: Ankita Thakur

First published: December 2025

Production reference: 1311225

Published by Packt Publishing Ltd.
Grosvenor House
11 St Paul's Square
Birmingham
B3 1RB, UK.

ISBN 978-1-83763-286-2
www.packtpub.com

Foreword

To the individual who finds themselves perpetually refreshing their cybersecurity news feed with a mix of dread and morbid fascination: welcome. You hold in your hands not just a book, but a carefully curated compilation designed to illuminate the ever-shifting landscape of Microsoft Security. If your current approach to cyber defense feels like playing digital whack-a-mole blindfolded, then congratulations, this is the guide for you.

The very mention of "cybersecurity" can conjure images of complex diagrams that rival abstract art and acronyms that multiply faster than rabbits. But rest assured, while this volume does indeed contain its fair share of intricate concepts and an alphabet soup of technical terms, the mission is to render them approachable. Consider this book your professional navigation through the intricacies of Microsoft Defender XDR, the strategic insights of Microsoft Sentinel, and the groundbreaking capabilities of Microsoft Security Copilot.

Whether you're seeking insight, practical solutions, or a new viewpoint on AI and cybersecurity, this book aims to provide useful and varied advice. Some content may touch on marketing concepts to offer a complete picture of Microsoft's security tools, but the goal is accessibility and value for all readers.

This work focuses on aligning your organization's security strategies with best practices, highlighting the importance of adopting DevSecOps principles. And no book like this would be complete without the discussing the role of Artificial Intelligence and Microsoft's Security Copilot in enabling proactive threat management and providing actionable insights, helping teams close skills gaps through advanced engineering solutions.

This book serves as a technical resource aimed at improving security posture and supporting the development of organizational resilience. It discusses the ongoing nature of cybersecurity, noting that best practices evolve frequently. The book covers a range of Microsoft solutions to provide guidance on protecting business assets.

So, arm yourself with knowledge (and perhaps a strong coffee), and prepare to transform your organization's security strategy. The digital realm is continuously evolving, and with the insights contained herein, you'll not only be prepared but truly leading the charge.

Rod Trent

Author of fiction and technical books, Microsoft Senior Product Manager for Security and AI

Contributors

About the authors

Jose Lazaro is a partner technical manager at Microsoft, where he leads a team of architects delivering next-generation security and AI business solutions. He works closely with partners and customers to design scalable, secure architectures that support modern cloud and hybrid environments.

Before joining Microsoft, Jose held senior leadership roles, including head of security and vice president of consulting at leading UK managed security service providers. He brings deep hands-on experience across SIEM and XDR platforms, with a career focus on designing, deploying, and operating enterprise security solutions at scale. His technical contributions and community leadership led to his recognition as a Microsoft MVP for SIEM and XDR.

Jose has worked with Microsoft security technologies for over a decade and is deeply passionate about cybersecurity, cloud adoption, and enabling digital transformation while maintaining a strong and practical security foundation.

Outside of work, Jose is an avid cyclist who enjoys exploring some of the most beautiful landscapes in the UK on two wheels.

Marcus Burnap (CISSP, Microsoft MVP, MCT) is a cybersecurity leader and Microsoft security specialist with expertise in SIEM, **extended detection and response** (**XDR**), and AI-enabled security operations. His experience includes designing and deploying Microsoft Sentinel and Defender XDR at enterprise scale, guiding organizations through security transformation programs, and embedding AI-driven detection and response with Microsoft Security Copilot. He is recognized as a Microsoft MVP for his contributions to the global security community, including scenario-based consulting, thought leadership, and event speaking. Marcus is also a contributing voice in advancing SecDevOps practices and AI adoption in modern security operations.

About the reviewers

Sami M'Barek is a Nordic-based senior security architect working with advanced SOC, SIEM, and XDR platforms. He designs multi-tenant Sentinel environments, GDAP/Lighthouse models, and automated detection pipelines for MSSPs. His work includes several in-house tools for content orchestration, incident classification, and ML-assisted threat detection.

Sami Lamppu is a principal cloud security lead at Elisa and a Microsoft security dual MVP with extensive industry experience. He has a strong background in enterprise security that has expanded to include cloud security, SIEM, and modern SOC practices. He specializes in strategic security advisory, helping organizations strengthen their cloud posture and adopt AI-assisted operations with Microsoft security solutions.

He is the co-author of several books, including the *Entra ID Attack and Defense Playbook*, and is a frequent speaker at industry events on cloud security and modern SOC practices.

Ru Campbell is a Microsoft MVP, Packt author, LinkedIn Learning instructor, and organizer of the free monthly Microsoft 365 Security & Compliance User Group. A regular conference speaker, he is head of Microsoft security at Threatscape, where he oversees the consultancy practice and leads Overwatch, the company's Microsoft-centric security posture management platform and service.

Table of Contents

Preface **xxiii**

Free Benefits with Your Book ..... **xxix**

Chapter 1: Microsoft Defender XDR in the Age of Security Copilot **1**

Free Benefits with Your Book • 2

Introduction to XDR: Embracing the future of cybersecurity **2**

Understanding XDR • 2

The role of AI in transforming the modern security landscape • 3

Overview of Microsoft's unified security approach and ecosystem • 4

The initiation of Security Copilot: AI-powered threat detection and response • 6

Future outlook • 7

Aligning with cybersecurity best practices **8**

Implementing a comprehensive cybersecurity framework • 8

Top 10 critical security controls for effective threat mitigation • 9

Adapting best practices to your organization's unique needs • 10

The CIA triad: A cornerstone of cybersecurity • 12

Applying the CIA triad to your security strategy • 14

Assessing your current security posture and the need for modern DevSecOps **17**

Conducting a security assessment and gap analysis • 17

Security assessment • 17

Gap analysis and benchmarking • 20

Identifying and prioritizing security risks and vulnerabilities • 21

Vulnerability management • 22

Building a strong vulnerability management program with Microsoft solutions • 23

Vulnerability assessment with Microsoft Defender Vulnerability Management • 24

Prioritizing vulnerabilities with Microsoft Secure Score • 24

Establishing a continuous vulnerability management program • 25

Creating an effective remediation process • 26

Developing a strategic roadmap for security enhancements with the Microsoft security stack • 28

Identity management: The first line of defense • 29

Embracing Zero Trust architecture • 29

Securing user endpoints • 29

Modern SecOps for a swift response • 29

Protecting data assets • 30

Securing applications against threats • 30

Infrastructure security for hybrid environments • 30

Enter DevOps and DevSecOps • 31

Integrating DevSecOps practices • 31

Key components of DevSecOps • 32

DevSecOps and Microsoft tools and services • 33

Embracing DevSecOps in your organization • 34

The Microsoft Security competitive advantage and Security Copilot 37

The future of AI and Microsoft in cybersecurity • 38

Summary 39

Chapter 2: Aligning with Cybersecurity Frameworks 41

Essential cybersecurity frameworks 42

National Institute of Standards and Technology Framework • 42

The five pillars of the NIST CSF • 42

Getting started with NIST CSF 2.0 • 44

NCSC Cyber Assessment Framework (CAF) – United Kingdom • 44

Benefits of NCSC CAF • 46

Adoption and implementation of NCSC CAF • 46

Cyber Essentials and Cyber Essentials Plus – United Kingdom • 47

Cyber Essentials • 47

Cyber Essentials Plus • 48

ISO/IEC 27001 – The international gold standard for information security • 49

Controls in ISO/IEC 27001 • 49

ISO/IEC 27001:2022 Annex A – overview of controls • 50

The 10 Steps to cybersecurity • 50

General Data Protection Regulation (GDPR) • 52

The GDPR's implications for cybersecurity • 52

SOC 2 • 54

The Center of Internet Security (CIS) • 55

Core components of CIS – Controls and Benchmarks • 55

CIS and Microsoft • 56

Automated security assessments • 56

Securing Azure with the CIS Benchmarks • 56

What framework should you align your organization to? • 57

The Secure Future Initiative (SFI) – a new foundation for trust 58

Microsoft's CAF and Azure's WAF 60

CAF • 60

WAF • 62

CAF versus WAF – which one should you use? • 64

Defining and implementing zero-trust strategies for your infrastructure 64

The emergence of Zero Trust – a modern cybersecurity model • 64

From reactive to proactive – the shift in cyber defense • 65

Core principles of Zero Trust • 66

Never trust, always verify • 66

Least privilege access • 66

Assume breach • 66

Micro-segmentation • 67

Explicit verification • 67

Implementing a Zero-Trust architecture • 67

Identity verification • 67

Device security • 68

Network security • 68

Data security • 68

Security policies and governance • 68

Awareness and training • 69

Challenges and considerations in zero-trust implementation • 69

Legacy systems • 69

Cultural shift • 69

Cost and complexity • 70

Integration with existing security measures • 70

Regulatory compliance • 70

Planning your Zero-Trust strategy • 70

Understanding your environment • 71

Exploring Zero-Trust technologies and solutions 73

Identity and Access Management (IAM) • 73

Network security • 74

Data security and encryption • 74

Security policies and governance • 74

How can you achieve Zero Trust with Microsoft solutions? • 75

The future of Zero Trust • 76

Summary 77

Chapter 3: Microsoft Security for Partners 79

The Microsoft AI Cloud Partner Program 80

Benefits of joining the Microsoft AI Cloud Partner Program for security providers • 80

Becoming a member of the Microsoft AI program • 81

Becoming a Microsoft Solutions Partner • 82

Solutions Partner for Security • 83

Solution designation for SMB • 84

Strategic value • 86

Security advanced specializations • 86

Exploring security-focused partner programs 88

Microsoft-certified MXDR partner program • 88

Why should a security partner become a Microsoft-certified MXDR provider? • 88

Qualification process for Microsoft-certified MXDR solutions • 89

Requirements for verified MXDR solutions • 89

Technical standards for verification • 90

Microsoft Intelligent Security Association • 91

MISA membership pathway • 92

Microsoft Cybersecurity Investment Program 92

Partner benefits and business impact • 93

Understanding the relationship between CSI and MCI • 93

Microsoft Sentinel Migration and Modernization Program • 94

Core components of the program • 94

Microsoft FastTrack partners • 96

What do FastTrack partners deliver? • 96

Partner security: protecting your customers 97

Mandatory security requirements for Microsoft partners • 97

Customer access models: from DAP to GDAP • 98

DAP • 98

GDAP • 98

Azure fraud notifications • 100

Get recognized by Microsoft for your success: CPOR and PAL 101

CPOR • 101

How to claim CPOR • 102

PAL: get recognized for Azure impact • 102

Getting PAL set up • 103

PAL setup • 104

Azure Lighthouse deployment • 104

Summary 105

Chapter 4: Core Foundations: Initial Setup, Strategic Financial Planning, and Business Insights 107

Foundational requirements: Setting up Microsoft 365 tenants and Azure 108

What is a Microsoft 365 tenant? • 108

Configuring your Microsoft 365 tenant for enhanced security • 109

Identity and access management (IAM) • 113

Microsoft Entra ID security defaults • 113

Multitenant organizations in Microsoft 365 • 115

Azure subscriptions • 116

Understanding and choosing subscription models • 116

Azure subscription best practices • 117

Structuring your Azure subscriptions • 117

Choosing your workspace architecture for Microsoft Sentinel: Centralized, decentralized, or hybrid 118

Centralized workspace architecture • 118

Design considerations and trade-offs • 118

When should you choose a centralized architecture? • 119

Decentralized workspace architecture • 119

Design considerations and trade-offs • 120

When should you choose a decentralized model? • 120

Hybrid workspace architecture • 121

Design considerations and trade-offs • 122

Challenges and strategic planning • 122

When should you choose a hybrid architecture? • 123

Cost, access control, and scalability • 124

Cost: Visibility versus duplication • 124

Access control: Simplicity versus precision • 124

Scalability: One size or fit for purpose? • 125

Making the right call • 125

Designing role-based access control (RBAC) for Microsoft Sentinel 126

The role model: How access works in Sentinel • 126

Practical application: A tiered SOC model • 127

Role assignment pitfalls to avoid • 128

Building an RBAC life cycle: From assignment to expiry • 128

Managing access at scale with Azure Lighthouse • 129

When should you use custom roles? • 129

Segment access by scope • 129

Audit, rotate, review • 130

Sentinel pre-deployment checklist: Planning with precision 130

Step 1: Identifying and prioritizing use cases • 131

Step 2: Estimating ingestion and budget • 131

Step 3: Selecting your workspace architecture • 132

Step 4: Assembling your deployment team • 132

Step 5: Planning Defender and Microsoft 365 integration • 132

Step 6: Defining your automation strategy • 133

Step 7: Documenting governance, roles, and maintenance • 135

Workspace naming conventions • 136

Dedicated cluster guidance for high-volume ingestion • 136

Data connector planning in Microsoft Sentinel • 138

Strategic automation in Microsoft Sentinel • 140

Automation rules: Centralizing incident flow • 141

Playbooks: Orchestrating response with Logic Apps • 141

Use cases for Sentinel automation • 142

Monitoring the health of your automations • 142

Multitenant management with Defender XDR and Microsoft MTOs 143

Defender XDR: Centralized security across tenants • 143

Important considerations and limitations • 144

Workspace Manager for MSSPs • 145

Deployment roadmaps: Migration and modernization of your SIEM solution 147

Greenfield deployment • 147

Brownfield migration • 148

Side-by-side strategy • 149

Strategic considerations for all migration paths • 150

Phase-based migration framework • 150

Phase 1: Discovery • 150

Phase 2: Design • 151

Phase 3: Implementation • 151

Phase 4: Operationalization • 151

Microsoft Sentinel Migration and Modernization Program (MSMMP) • 152

Best practices for migrating from legacy SIEMs • 152

Essential tools and resources • 153

Data retention and long-term storage 154

Optimizing cost, coverage, and performance in Microsoft Sentinel • 156

Filtering, controls, and budget protection • 156

Financial journey: Assessing costs from groundwork to full operation 158

Microsoft Sentinel's pricing model • 158

What you pay for in Microsoft Sentinel • 158

Sentinel data tiers and usage model • 159

Analytics tier: High fidelity, high value • 159

Sentinel data lake: Cost-optimized secure data storage • 159

Important shift from older models • 160

Summary 161

Chapter 5: Getting Ready for Action: Assess Your Microsoft 365 Tenant 163

Technical requirements 163

Getting ready for action: Assessing your Microsoft tenant with ScubaGear 164

License requirements • 164

Getting ScubaGear up and running • 164

Step 1: Downloading the ScubaGear script and preparing your computer • 164

Step 2: Setting up PowerShell to run ScubaGear • 167

Step 3: Running ScubaGear • 169

Step 4: ScubaGear output • 172

Automating tenant monitoring with Maester and Azure DevOps 173

Setting up the Maester repository in Azure DevOps • 174

Adding a client's location • 177

Setting up the Azure pipeline • 182

Creating an Entra ID application registration • 182

Granting permissions to Exchange Online (optional but recommended) • 189

Granting permissions to Azure (optional) • 189

Creating a client secret • 191

Creating an Azure pipeline • 192

Zero Trust Assessment Tool guide 197

Prerequisites • 197

First-time module installation • 199

Telemetry settings (optional) • 203

Feeding Security Copilot with real context: Maester test results • 204

Summary 205

Chapter 6: Foundational Architecture and Secure Access for Sentinel Deployments for MSSPs 207

Designing secure and scalable Sentinel foundations using Azure subscriptions and management groups 208

Management groups and log collection • 208

Management groups and their impact on log collection • 209

Choosing the right subscription: Existing versus new • 209

Creating a new management group • 210

Creating a new subscription • 212

RBAC and unified XDR access • 212

Key scenarios where Azure Lighthouse or B2B remains essential • 214

Building structured, least privilege access with RBAC, Entra ID, and PIM 215

Establishing foundational access controls • 215

Overall URBAC concept of least privilege • 216

Scenario: Team-based role design for Sentinel operations • 218

Deploying foundational RBAC artifacts – home/MSSP tenant 219

Creating security groups • 219

Creating an access package • 221

Defining role-based access groups • 228

Enabling PIM • 229

What about roles? • 235

Defining the right roles for unified control 238

Setting up RBAC/Azure RBAC • 239

Example role matrix for home tenant groups • 239

Common role-based requirements • 240

Moving forward: Multitenant role assignment and delegated access • 241

Choosing the right integration method: GDAP, Lighthouse, or B2B? • 241

Get This Book's PDF Version and Exclusive Extras • 242

Summary 242

Chapter 7: Customer Onboarding: The Power of Azure Lighthouse 243

Technical requirements 244

Azure Lighthouse onboarding and best practices 244

Establishing cross-tenant access via Azure Lighthouse • 245

Reviewing the ARM template • 245

Contextual usage and best practices • 248

Deploying an Azure Lighthouse template • 249

Granular delegated admin privileges (GDAP) connection 253

Step 1: Creating a new admin relationship • 253

Step 2: Sending and finalizing the GDAP request • 256

Step 3: Mapping GDAP roles to Azure AD security groups • 257

Step 4: Accessing the client environment via GDAP • 259

Entra ID B2B access model 259

Mandatory requirements • 261

Client-side onboarding for B2B access 261

Step 1: Configuring external identity and cross-tenant access • 261

Step 2: Creating an authentication context and Conditional Access policy • 264

Step 3: Inviting initial B2B MSSP sponsors • 265

Step 4: Adding the home tenant as a connected organization • 266

Step 5: Creating access package security groups • 269

Step 6: Adding role assignments to the groups • 274

Azure/Microsoft Sentinel role assignments • 274

Step 1: Making roles available via the authentication context • 276

Step 2: Repeating the process for required roles • 278

Step 3: Assigning active roles • 279

Step 7: URBAC Defender roles • 282

Step 8: External access governance setup • 286

Step 9: Conditional Access authentication context • 291

Outcome: Bringing external access experience together • 294

Summary 298

Chapter 8: Joining the Dots: Microsoft Sentinel Deployment Plan and Features 301

Permissions in the Unified Microsoft Defender Portal 302

Microsoft Sentinel SIEM permissions remain unchanged • 302

Microsoft Defender XDR permissions are separate • 303

Minimum URBAC role for Defender XDR investigations • 303

Special workload considerations • 303

Managing multiple Sentinel workspaces • 304

Legacy RBAC and rollback safety • 305

Privileged Identity Management and auditing • 305

Microsoft Sentinel SIEM deployment plan 306

30-60-90 day deployment plan: from setup to security outcomes • 306

Days 0–30: Build the foundation – data collection and connector configuration • 306

Days 31–45: Optimize collection and enhance visibility • 308

Days 46–60: Deepen detection, investigation, and response • 308

Days 61–75: Innovate data collection and enable long-term insights • 310

Days 76-85: Strengthen detection with intelligence and analytics • 311

Days 86–90: Empower analysts with proactive tools • 312

Executing your deployment plan for Microsoft Sentinel 313

Manual deployment of LAWS and Microsoft Sentinel • 313

Creating a LAWS • 314

Enabling Microsoft Sentinel • 317

Core Sentinel initial setup settings • 318

Enabling diagnostic logs in your LAWS • 322

Connecting the unified platform 328

Optional: Multi-tenant management • 333

Collecting data and deploying solutions • 334

Deploying content hub solutions • 335

The backbone of Microsoft Sentinel 346

Introduction to KQL and detection rules • 346

Understanding KQL syntax basics • 347

Essential KQL functions you need to know • 348

Rule logic and entity mapping • 351

Entity mapping • 353

Introduction to Summary rules • 364

Creating a summary rule • 365

Deploying watchlists • 367

Introduction to Automation • 370

Logic Apps • 371

Introduction to Incidents • 375

Introduction to Workbooks • 378

Introduction to Hunting • 380

Advanced Hunting in Defender XDR • 382

Introduction to Threat intelligence 383

Introduction to the MITRE ATT&CK framework • 386

Summary 389

Chapter 9: Microsoft Sentinel Fast-Track Deployment Options and CI/CD Pipelines 391

Deploying and managing Microsoft Sentinel artifacts at scale 392

Deploying content via repositories • 392

Getting set up to deploy content via repositories • 392

Connecting your repository • 393

Connecting via a client tenant? • 394

Adding your repo details • 396

Repository setup • 397

Directory versus branch-based deployment models • 398

Client-specific pipelines and solution-driven deployment from a central repository • 398

Amending your pipeline file location • 405

Tracking deployment status • 406

Deploying content via Workspace Manager 408

Organizing with groups • 410

Publishing content • 411

Fast-track your IAC Sentinel deployment 413

Start with infrastructure, not content • 413

Sentinel deployment via code • 413

Manual (All-in-One v2) • 414

Sentinel deployment via Bicep • 416

Setting up the project in Azure DevOps • 416

Understanding the pipeline workflow • 434

The PowerShell script: Set SentinelContent.ps1 • 434

Deployment via Terraform • 435

Azure DevOps pipeline integration • 435

State management strategy • 436

Testing and validation • 436

Terraform automation and deployment architecture for multi-tenant Sentinel environments • 436

Azure DevOps pipeline execution 437

Azure DevOps: project setup 440

Deploying Sentinel with Terraform: Key Vault and sub-tenant setup 440

Adding backend secrets • 441

Creating a new service connection in Azure DevOps • 442

Creating a storage account • 442

Creating a client blob container in the storage account • 443

Sentinel Terraform sub-tenant setup • 443

Creating a client blob container • 443

Preparing a client variables file • 443

Executing the pipeline • 444

Summary 445

Chapter 10: Activate, Configure, Deploy: Your First Steps with Security Copilot 447

Technical requirements 447

Foundational requirements: Azure subscription and SCUs • 448

Understanding SCUs: provisioned versus overage • 449

Authentication and delegated access model 449

Role-based access in Security Copilot • 450

Plugin permissions and the least privilege model • 450

Cross-tenant collaboration and shared sessions • 451

Onboarding process: getting Security Copilot up and running 451

Step 1: Provisioning your capacity • 452

Option 1: Provisioning via the Security Copilot portal • 452

Option 2: Using the Azure portal • 456

Step 2: Configuring your default environment • 458

Offboarding: if you need to step away • 462

Key topics around Microsoft Security Copilot and AI integration 464

MCP server inside Copilot Studio • 464

Security Copilot for MSSPs: current limitations and future potential • 464

Access, capability, and model improvements • 465

Protecting your IP as an MSP • 465

Secure exchange: agents and the security marketplace • 465

Security agents via Visual Studio dev packages • 466

Future outlook: Microsoft Sentinel and Defender XDR 466

A more unified console: Sentinel inside Defender XDR • 467

Entering the Fabric-based security data lake • 467

Changing where and how data is stored and processed • 467

What sales and security teams should be saying • 468

Get This Book's PDF Version and Exclusive Extras • 470

Chapter 11: Unlock Your Exclusive Benefits 471

Unlock this Book's Free Benefits in 3 Easy Steps 471

Other Books You May Enjoy 477

Index 479

Preface

Unified security operations, more commonly known as unified SecOps, is the practice of running prevention, detection, response, and governance as one continuous operating model. It brings identity, endpoint, email, cloud, and application telemetry into a single investigative workflow, and it standardizes how teams triage, contain, recover, and learn across the enterprise. In *Unified SecOps Playbook: End-to-end enterprise security with Microsoft Sentinel, Defender XDR, and Security Copilot*, Jose Lazaro and Marcus Burnap show how to apply this approach in the Microsoft ecosystem by combining Sentinel, Defender XDR, and Security Copilot into an end-to-end SecOps platform.

This playbook is a practical guide to modern cybersecurity using Microsoft tools. It provides step-by-step guidance for set-up and deployment, integrates Zero Trust with AI-assisted operations, and aligns common engineering practices with established security frameworks. Throughout the book, we focus on repeatable deployment patterns that scale across large, complex environments, while staying grounded in the realities of cost, governance, and operational maturity.

Key topics covered in this book include the following:

- Deploying Microsoft Sentinel with end-to-end guidance, including CI/CD automation and scalable architecture
- Integrating Security Copilot with Defender XDR to enable AI-driven threat detection and response
- Implementing Zero Trust strategies and aligning with modern DevSecOps and cybersecurity frameworks

We start by grounding you in core XDR concepts, security frameworks, and the competitive advantages of the Microsoft security ecosystem. From there, we move into the foundational work required to deploy Microsoft Sentinel in a way that is secure, governable, and ready for enterprise operations.

You will learn how to configure the security infrastructure that underpins an effective SOC, including **role-based access control** (**RBAC**), identity governance with Microsoft Entra ID and **privileged identity management** (**PIM**), and the architectural decisions that make deployments repeatable across tenants, subscriptions, and business units.

Once this foundation has been established, we introduce practical operating workflows: how to onboard and normalize data, how to design detections and incident handling, how to introduce automation safely, and how to turn hunting into an operational discipline. We also cover strategic financial planning and cost management so that capability growth does not come with uncontrolled ingestion spend.

Advanced topics are covered, including integrating Zero Trust strategies, modern DevSecOps practices, and Microsoft Partner programs to support fast-moving delivery models. You will also explore fast-track options and CI/CD pipelines to accelerate rollouts while maintaining change control and auditability.

Finally, we bring AI into the SecOps workflow. By integrating Security Copilot with Defender XDR and Microsoft Sentinel, you will see how generative AI can accelerate investigation and response without sacrificing governance.

By the end of this book, you will have gained practical, field-tested insights into security automation, threat detection, and AI integration. You will be able to implement and manage Microsoft Defender XDR and Microsoft Sentinel in complex environments, and you will have the confidence to drive scalable and secure solutions across your organization or customers.

You will learn how to do the following:

- Deploy Microsoft Sentinel with RBAC, governance, and financial planning
- Align security programs with Zero Trust and global frameworks
- Automate assessments with ScubaGear, Maester, and Azure DevOps
- Build secure Sentinel foundations with Entra ID, PIM, and management groups
- Onboard customers and partners via Azure Lighthouse and GDAP
- Master the Sentinel workflow backbone: KQL, detections, automation, and MITRE ATT&CK
- Accelerate rollouts using FastTrack deployment and CI/CD pipelines
- Harness Security Copilot for AI-assisted detection and investigation

In writing this book, we wanted to share the real-world patterns, decision points, and operational guardrails that separate a functional deployment from a SOC-ready capability.

Who this book is for

This book is for you if you have a foundational understanding of Microsoft security products and Azure and are now looking to expand your expertise by incorporating Microsoft Sentinel. Security experts using alternative SIEM tools who want to adopt Microsoft Sentinel as an additional or replacement service will also find this book useful.

What this book covers

Chapter 1, *Microsoft Defender XDR in the Age of Security Copilot*, introduces the modern XDR landscape and how Microsoft is reshaping detection and response with unified telemetry and AI-assisted workflows.

Chapter 2, *Aligning with Cybersecurity Frameworks*, shows how to map SecOps goals and technical controls to widely adopted frameworks and how to use those frameworks to guide design and governance decisions.

Chapter 3, *Microsoft Security for Partners*, explores partner delivery models, multi-tenant considerations, and the practical mechanics of operating across customers using approaches such as Lighthouse and GDAP.

Chapter 4, *Core Foundations: Initial Setup, Strategic Financial Planning, and Business Insights*, covers the early-stage decisions that determine long-term success, including architecture patterns, licensing and cost drivers, and foundational governance.

Chapter 5, *Getting Ready for Action: Assess Your Microsoft 365 Tenant*, provides a structured approach to baselining identity, endpoint, and SaaS security posture so your Sentinel and Defender deployment is anchored to measurable risk reduction.

Chapter 6, *Foundational Architecture and Secure Access for Sentinel Deployments for MSSPs*, details secure access models, RBAC, PIM, management group strategy, and tenant/subscription design patterns for scalable operations.

Chapter 7, *Customer Onboarding: The Power of Azure Lighthouse*, walks through onboarding workflows and the operational guardrails that enable secure, auditable, and efficient service delivery at scale.

Chapter 8, *Joining the Dots: Microsoft Sentinel Deployment Plan and Features*, delivers a practical, phased deployment approach that turns Sentinel from installed to operational, covering ingestion, detections, triage, automation, and hunting.

Chapter 9, *Microsoft Sentinel Fast-Track Deployment Options and CI/CD Pipelines*, explains how to accelerate deployments with repeatable automation, including infrastructure-as-code patterns and pipeline-based content deployment.

Chapter 10, *Activate, Configure, Deploy: Your First Steps with Security Copilot*, shows how to onboard Security Copilot, integrate it with Defender XDR and Sentinel, and apply governance so AI accelerates outcomes without introducing new risk.

To get the most out of this book

The following are recommended to get the most out of this book:

- A working knowledge of Microsoft Defender XDR, Microsoft 365 security, and core Azure concepts such as subscriptions, resource groups, and RBAC
- Familiarity with SOC processes for triage, investigation, and incident response (for example, how to interpret alerts, evidence, and timelines)
- Basic comfort with reading and adapting queries and scripts, including KQL, PowerShell, and YAML (you do not need to be a developer)
- Access to an Azure subscription and a Microsoft 365 tenant where you can evaluate Sentinel, Defender XDR, and (if available) Security Copilot in a lab or controlled environment
- Tooling such as Azure CLI, PowerShell, Git, and either GitHub or Azure DevOps for the automation and CI/CD examples

Download the example code files

The code bundle for the book is hosted on GitHub at `https://github.com/PacktPublishing/Unified-SecOps-Playbook`. We also have other code bundles from our rich catalog of books and videos available at `https://github.com/PacktPublishing`. Check them out!

Download the color images

We also provide a PDF file that has color images of the screenshots/diagrams used in this book. You can download it here: `https://packt.link/gbp/9781837632862`.

> This book contains extended images and screenshots that may affect readability in certain formats or print sizes. These images are included for illustrative purposes to support conceptual understanding. Please note that the examples, visuals, and accompanying content may vary or be updated based on your input, use cases, or contextual requirements.

Conventions used

There are a number of text conventions used throughout this book.

`CodeInText`: Indicates code words in text, data tables, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and resource identifiers. For example: "You can control this using the `-EnableTelemetry` switch."

A block of code is set as follows:

```
SecurityAlert
| where ProviderName == "Microsoft Defender for Endpoint"
| project TimeGenerated, AlertName, Severity, CompromisedEntity
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
steps:
- task: PowerShell@2
  inputs:
    script: |
      pwsh ./pipelines/deploy-sentinel.ps1 -Environment "Prod"
-WorkspaceId <WORKSPACE_ID>
```

Any command-line input or output is written as follows:

```
az sentinel analytics-rule list --resource-group <RG> --workspace-name
<LAW>
```

Bold: Indicates a new term, an important word, or words that you see on the screen. For instance, words in menus or dialog boxes appear in the text like this. For example: "Select **Incidents** from the Microsoft Sentinel navigation."



Warnings or important notes appear like this.



Tips and tricks appear like this.

Get in touch

Feedback from our readers is always welcome.

General feedback: If you have questions about any aspect of this book, please email us at `customercare@packt.com` and mention the book title in the subject of your message.

Errata: If you find an error, please visit `http://www.packt.com/submit-errata`, click **Submit Errata**, and complete the form.

Piracy: If you find illegal copies online, please email `copyright@packt.com` with the location details.

If you are interested in becoming an author, please visit `http://authors.packt.com/`.

Share your thoughts

Once you've read *Unified SecOps Playbook*, we'd love to hear your thoughts! Please click here to go straight to the Amazon review page for this book and share your feedback.

Your review is important to us and the tech community and will help us make sure we're delivering excellent quality content.

Free Benefits with Your Book

This book comes with free benefits to support your learning. Activate them now for instant access (see the "*How to Unlock*" section for instructions).

Here's a quick overview of what you can instantly unlock with your purchase:

PDF and ePub Copies



- Access a DRM-free PDF copy of this book to read anywhere, on any device.
- Use a DRM-free ePub version with your favorite e-reader.

Next-Gen Web-Based Reader



- **Multi-device progress sync**: Pick up where you left off, on any device.
- **Highlighting and notetaking**: Capture ideas and turn reading into lasting knowledge.
- **Bookmarking**: Save and revisit key sections whenever you need them.
- **Dark mode**: Reduce eye strain by switching to dark or sepia themes.

How to Unlock

UNLOCK NOW

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.



***Note**: Keep your invoice handy. Purchases made directly from Packt don't require one.*

1

Microsoft Defender XDR in the Age of Security Copilot

Welcome to the first chapter of this book on Microsoft Security. We would like to express our gratitude for choosing to read this book! Whether you're new to the topic or are in search of solutions and fresh perspectives to tackle challenges in cybersecurity, this book delivers diverse insights and practical advice on navigating **Artificial Intelligence** (**AI**) and Copilot within the Microsoft Security ecosystem.

In this chapter, we'll begin with a concise review of fundamental concepts, including Defender XDR, the Microsoft unified **Security Operations** (**SecOps**) platform, Microsoft Sentinel, and Security Copilot. It's important to understand that Microsoft Defender XDR (an **Extended Detection and Response** (**XDR**) solution) and Microsoft Sentinel (a SIEM platform) are complementary. While XDR provides cross-domain threat detection and response across identities, devices, and cloud apps, Sentinel extends this visibility by acting as a centralized hub for event management, analytics, and long-term threat intelligence.

The initial overview is designed for those who may not yet have deep security expertise, such as junior IT engineers, early-career software developers, business-risk analysts, or project coordinators involved in security-related work. It establishes a clear foundation by explaining the meaning and relevance of commonly used security terms and buzzwords, such as SIEM, SOAR, XDR, EDR, and UEBA, ensuring readers share a consistent baseline understanding before moving into more advanced material.

Admittedly, some sections in this chapter will lean into marketecture, the high-level, marketing-style view of technology, which is also a necessary evil and long-term annoyance of deep technical readers. While not purely technical, this perspective is useful, as it provides broader context on Microsoft's security toolset and its overall capabilities before we drill into deeper, hands-on details.

In this chapter, we will set the scene, contextualize, and explore the following topics:

- Introduction to XDR: Embracing the future of cybersecurity
- Aligning with cybersecurity best practices
- Applied fundamentals of your current security stance and the need for modern DevSecOps
- The Microsoft Security competitive advantage and Security Copilot

Free Benefits with Your Book

Your purchase includes a free PDF copy of this book along with other exclusive benefits. Check the *Free Benefits with Your Book* section in the Preface to unlock them instantly and maximize your learning experience.

Introduction to XDR: Embracing the future of cybersecurity

As digital technology constantly evolves, companies are confronted with increasingly complex and relentless cyberthreats. The conventional strategies for cyber defense are less able to adapt to this constant evolution. This challenge has paved the way for the emergence of XDR, a groundbreaking shift in cybersecurity. XDR offers an all-encompassing strategy for identifying, investigating, and countering cyber threats, moving beyond the constraints of isolated security measures.

Understanding XDR

At its core, XDR is a holistic security approach that integrates and correlates data from multiple security layers, including endpoints, networks, cloud platforms, and applications. By unifying data and applying advanced analytics and **Machine Learning** (**ML**) algorithms, XDR platforms enable security teams to gain a comprehensive view of their managed environment's security posture, allowing them to detect, investigate, and respond to threats with unparalleled speed and accuracy.

XDR platforms collect and analyze telemetry data from across the enterprise, utilizing sophisticated algorithms to identify patterns, anomalies, and indicators of compromise that might otherwise go unnoticed. This data-driven approach allows XDR to detect threats, such as **Advanced Persistent Threats (APTs)**, zero-day exploits, and fileless malware, that traditional, single-focused security tools that only view siloed or platform-specific data, such as network intrusion detection or standalone endpoint protection, might miss.

Moreover, XDR platforms provide security teams with contextualized, actionable insights into threats by leveraging a wide range of telemetry, enabling them to prioritize their efforts and focus on the most critical incidents. By correlating alerts and events across multiple security layers, XDR helps security analysts connect the dots between seemingly disparate incidents, uncovering the full scope of complex, multi-stage attacks.

In the context of Microsoft's security solutions, **Microsoft Defender XDR** offers an XDR capability that integrates signals from various sources, such as **Microsoft Defender for Endpoint (MDE)**, Microsoft Defender for Office 365, Microsoft Defender for Identity, and Microsoft Defender for Cloud Apps. The Microsoft Defender XDR platform is the main portal you visit when using the suite, with licensed features then populated into the page. It acts as a centralized security center, correlating and analyzing signals from across Microsoft's security products to detect, investigate, and respond.

This single-pane-of-glass view and integration allows security teams to automate responses and remediate threats across their entire digital estate, enhancing both efficiency and effectiveness in their SecOps. Microsoft Defender XDR is no longer confined to Microsoft environments. It supports telemetry from multi-cloud, on-premises, and **operational technology (OT)** systems via integrations with Microsoft Defender for Cloud and Azure Arc.

Now that we have discussed the central security brain, let's understand how AI is already leveraged and its future in security.

The role of AI in transforming the modern security landscape

AI and ML are key elements of XDR platforms, as they enable the real-time analysis of massive volumes of security data, identifying threats that might evade traditional rule-based detection methods.

AI-powered XDR platforms can learn from historical data, adapting to new threats as they emerge and continuously refine their detection algorithms to improve accuracy and reduce false positives. There are a whole host of options currently on the market in 2025, such as Cisco XDR, CrowdStrike Falcon, and SentinelOne Singularity. The adaptive, self-learning approach of these products is critical for keeping pace with the rapidly evolving threat landscape, where new attack vectors and techniques emerge daily.

Furthermore, AI and ML enable XDR platforms to automate and orchestrate key SecOps by analyzing historical data, learning behavioral patterns, and then requesting human input when needed to fine-tune or validate their decision-making logic.

An XDR platform may detect unusual lateral movement behavior on an endpoint and compare it to past incidents. If the system is confident and resembles known malicious activity, it may automatically isolate the device. However, if the behavior is borderline, for example, a software developer running new administrative scripts, the XDR system could refer to predefined weighting rules or could pause and ask the user or an analyst to confirm whether the activity is legitimate.

The analyst's decision is then fed back into the model, helping the platform improve accuracy for similar situations in the future. These operations range from prioritizing and investigating threats to responding to incidents and implementing remediation. By automating repetitive, time-consuming tasks, XDR frees up security teams to focus on more strategic initiatives, such as threat hunting and proactive defense.

Overview of Microsoft's unified security approach and ecosystem

Microsoft has been at the forefront of the XDR revolution, developing a comprehensive, unified security ecosystem that leverages the power of AI and ML to provide organizations with end-to-end threat protection. At the heart of Microsoft's approach is Microsoft Defender XDR, the cloud-native XDR platform that integrates and correlates security data from across the following Microsoft ecosystems:

- **Microsoft Defender for Endpoint**: Provides advanced **endpoint detection and response (EDR)** capabilities, using AI and ML to identify and block advanced threats in real time.
- **Microsoft Defender for Identity**: Leverages behavioral analytics and ML to detect and investigate identity-based threats, such as compromised credentials and insider attacks. With the new identity timeline in Microsoft Defender XDR, you now have a unified 30-day view of user activities, bringing together signals from Defender for Endpoint, Identity, and Cloud Apps to simplify investigations.

- **Microsoft Defender for Office 365**: Offers comprehensive protection against email-based threats, such as phishing and malware, using AI-powered threat intelligence and automated response capabilities.
- **Microsoft Defender for Cloud Apps**: Provides visibility and control over cloud applications and services, using AI and ML to detect and investigate suspicious activities and potential data breaches.
- **Microsoft Sentinel**: A cloud-native **Security Information and Event Management (SIEM)** platform that integrates with Microsoft Defender XDR, providing centralized security analytics, threat hunting, and incident response capabilities.

The idea behind integrating these solutions into a unified security suite, now called Microsoft Defender XDR, is to provide organizations with a complete view of their security posture, enabling them to detect, investigate, and respond to threats across the entire internal and external attack surface. The suite utilizes Microsoft's extensive threat intelligence network, which analyzes trillions of signals daily to identify new threats and provide actionable recommendations to security teams.

One of the key advantages of Microsoft's approach is its deep, seamless integration across the wider Microsoft ecosystem. This means organizations can immediately make use of the investments they already have in place, such as identity signals from Entra ID, device telemetry from Defender for Endpoint, email and collaboration insights from Microsoft 365, configuration controls from Intune, and cloud intelligence from Azure. Together, these components form a comprehensive, end-to-end security ecosystem without the need for additional agents, connectors, or standalone security tools. Additionally, Microsoft's commitment to open standards and interoperability enables Microsoft Defender XDR and Microsoft Sentinel to integrate with third-party security solutions. This flexibility allows organizations to build a best-of-the-breed security stack that meets their unique needs and requirements, enhancing their overall security posture without sacrificing existing investments.

So, with all these advancements and quick ways to gain immediate value, how has Microsoft presented its offering to the market?

The initiation of Security Copilot: AI-powered threat detection and response

Imagine a world where your security team isn't just reacting to cyber threats but proactively staying ahead, anticipating and neutralizing attacks before they can cause harm. Building upon the robust foundation of Microsoft Defender XDR, Microsoft has introduced a game-changer in cybersecurity: **Microsoft Security Copilot**. This AI-powered security assistant is poised to revolutionize the way security teams detect, investigate, and respond to cyber-attacks.

Security Copilot is now embedded within the Microsoft Defender portal, enabling seamless collaboration with Microsoft Sentinel. This integration allows analysts to triage, investigate, and respond to threats directly from a unified interface, in turn improving productivity, streamlining response workflows, and reducing context switching. Security Copilot brings real-time, contextual, and actionable insights to your team. For your organization, this means faster, more accurate threat detection and response, while also helping to close the cybersecurity skills gap by making it easier for junior analysts to contribute effectively from day one.

By analyzing security data from across the entire Microsoft Defender XDR suite and other integrated security tools, Security Copilot can spot potential threats. As of 2025, Security Copilot includes features such as the Phishing Triage Agent, which autonomously investigates user-reported phishing emails, resolving false positives and lightening the workload for security teams. It doesn't stop at detection; it offers clear recommendations for investigation and response and even automates key security tasks. For instance, if a suspicious email attachment surfaces, Security Copilot can automatically dissect and detonate the file, assess its potential impact, and hand over a detailed report to your security analyst.

This AI-driven approach doesn't just enhance efficiency; it transforms it. By significantly reducing the **Mean Time to Detect (MTTD)** and **Mean Time to Respond (MTTR)**, Security Copilot enables your security team to stay ahead of the ever-evolving threat landscape. This sped-up AI-powered process is what we call the **Machine Time to Respond (MTTR)**.

Perhaps one of the most remarkable features of Security Copilot is its natural language processing capabilities. With this, your team can engage with the platform using everyday language and straightforward questions in the form of prompts. Are you curious about the most critical threats facing your organization right now? Just ask. Want to know how to bolster your defenses against ransomware attacks? Security Copilot has got you covered. This accessibility not only reduces skill barriers but also accelerates decision-making, enabling your team to focus on what truly matters. Copilot's embedded intelligence not only answers prompts, but suggests follow-up questions, summarizes incidents with threat intel context, and adapts in real time to analyst behavior.

What's more, Security Copilot is always learning. It continuously absorbs insights from user interactions and integrates feedback to refine its responses over time. This adaptive learning means that as new threats emerge and tactics evolve, Security Copilot evolves right alongside them.

In essence, Microsoft Security Copilot adds to your security toolkit and fundamentally enhances it. It brings together state-of-the-art AI technology and deep security expertise, providing an intelligent assistant that works tirelessly to safeguard your organization. With Security Copilot, we're reshaping the way we think about defense.

Future outlook

Today, XDR, enhanced by AI and ML, is redefining how we defend against cyber threats. Traditional security measures are struggling to keep pace; analysts are fighting to find the right portal and data, but XDR aims to tackle this challenge. By delivering contextualized insights and automating essential security tasks, Security Copilot enables teams to operate more intelligently and efficiently in the face of cyber adversities.

Integrating XDR and AI-driven security solutions equips security teams with the tools to anticipate and counteract attackers effectively. Organizations adopting these solutions position themselves not just to handle current threats but to proactively prepare for future challenges.



Lessons from the field

You must enable Microsoft's unified SecOps platform as it seamlessly combines Microsoft Defender XDR, Microsoft Sentinel, and all the dependent connected services into one cohesive solution. You get a complete view of your security landscape from a single, intuitive dashboard. Imagine having all your alerts and data from every protected resource right at your fingertips, making it easier for your team to spot and tackle threats quickly and effectively.

Now that we've dug into XDR and seen how Microsoft's unified platform offers comprehensive coverage via its product stack for both cloud and on-premises infrastructure, let's dive into the next section. We'll explore how aligning with cybersecurity best practices ensures that the data in your XDR suite is accurate and genuinely valuable and aligned to your strategies and operational needs.

Aligning with cybersecurity best practices

Adopting a proactive, multi-layered approach to cybersecurity is isn't just about technology. It's about people, processes, and the very fabric of your organization. When you weave together these elements by empowering your team, refining your procedures, and embracing the right technologies, you enhance your organization's resilience against cyber threats. You're minimizing the risk of successful attacks while building a security culture that's adaptable, robust, and ready for whatever comes next.

Implementing a comprehensive cybersecurity framework

One of the key pillars of an effective cybersecurity strategy is adopting a comprehensive cybersecurity framework. These frameworks provide a structured approach to managing threats, ensuring that your critical assets and data are protected against a wide range of cyberattacks.

Consider **Microsoft's Cybersecurity Reference Architecture (MCRA)** as a prime example. It offers a holistic approach that spans the entire attack surface, helping organizations align their security practices with industry standards and best practices. The MCRA guides you in building a robust defense that's both effective and adaptable.

At the core of the MCRA is the concept of **Zero Trust**, which is a cybersecurity model that flips the traditional security mindset on its head. Instead of assuming that everything inside your network is trustworthy, Zero Trust operates on the principle that no network, device, or user should be trusted by default. Every access request must be verified and authenticated before being granted, regardless of location or device. In essence, everything that connects to your environment should be known and verified.

Zero Trust is built upon three fundamental principles:

- **Verify explicitly**: Don't take anything for granted. Every access request, whether it originates inside or outside your network, should be authenticated and authorized using all available data points. This includes user identity, location, device health, the service or workload being accessed, data classification, and any anomalies that might indicate unusual activity.
- **Use least-privileged access**: When it comes to access rights, less is more. Grant users only the permissions they need to perform their tasks, nothing extra. Implement strategies such as **Just in Time (JIT)** and **Just-Enough Access (JEA)**, along with risk-based adaptive policies and robust data protection measures. This ensures that you're securing both your data and maintaining productivity without unnecessary exposure.

- **Assume breach**: Hope for the best but prepare for the worst. Operate under the assumption that a breach has already occurred. Minimize potential damage by segmenting access based on networks, users, devices, and applications. Ensure all sessions are encrypted from end to end. Leverage analytics to gain visibility, detect threats, and continually strengthen your defenses.

Now, let's move on to what we've identified through some of our own experience as the top-priority security controls for effective threat mitigation.

Top 10 critical security controls for effective threat mitigation

In addition to adopting a comprehensive cybersecurity framework, it is recommended that organizations implement a set of security controls that are designed to mitigate the most common and damaging cyber threats. The **Center for Internet Security** (**CIS**) has identified and highlighted 18 critical security controls that organizations should prioritize in their cybersecurity efforts. Here, we will focus on the top 10 controls that are most relevant to Microsoft's security ecosystem:

- **Asset inventory and control**: Proactively manage all enterprise assets, including servers, endpoints, mobile devices, and IoT devices, to ensure that only authorized devices can access the internal network.
- **Software inventory and control**: Keep track of all software on the network, including operating systems, applications, and firmware, ensuring that only approved software can be installed and run on managed devices.
- **Vulnerability management**: Continuously identify, assess, manage, and remediate vulnerabilities that could impact your organization.
- **Administrative privileges**: Limit administrative privileges and enforce least-privilege principles. Ensure administrative accounts are used exclusively for admin tasks.
- **Secure configuration for assets and software**: Establish and maintain secure configurations for all managed assets, including servers, endpoints, and mobile devices.
- **Audit logs health**: Collect, manage, and analyze audit logs from key systems to detect, respond to, or recover from cyber incidents.
- **Email and web browser protection**: Email and browsers are common entry points for attackers. Control them to reduce the attack surface, for example, by using solutions such as **Microsoft Defender for Office 365** and web content filtering within MDE.

- **Malware defense plan**: Develop a comprehensive plan to prevent malicious code installation and execution. Effective patch management plays a key role here. Automate patching where possible and ensure you have runbooks ready to quickly address vulnerabilities.
- **Control of ports, protocols, and services**: Fully manage ports, protocols, and services on network devices. Use solutions such as JIT access to minimize attack windows.
- **Disaster recovery capabilities**: Ensure you have processes and tools in place to back up critical data. Regularly test your recovery plan to confirm it works in the event of an incident.



The latest version, **CIS Controls v8.1**, includes 18 critical security controls designed to help organizations manage risks effectively by focusing on foundational cyber hygiene, such as inventory management, secure configurations, and vulnerability management. These controls are applicable across various environments, including cloud and hybrid, and are a valuable resource for developing a comprehensive cybersecurity strategy.

For a detailed breakdown of the CIS Controls, you can visit the official CIS website here: `https://www.cisecurity.org/controls/cis-controls-list`.

Adapting best practices to your organization's unique needs

The Microsoft **Security Adoption Framework (SAF)** is designed to help organizations adapt their security practices to meet their unique requirements. Unlike static security frameworks, SAF is built to accommodate the specific needs and challenges each organization faces, enabling a tailored security approach that aligns with their risk profile.

SAF provides a structured way for organizations to assess their current security posture, identify gaps, prioritize initiatives, and develop a roadmap for security improvements. It emphasizes an end-to-end approach to security modernization, integrating with existing infrastructures and guiding organizations through key phases such as strategic planning, technical implementation, and ongoing security management. The framework also aligns with the MCRA, ensuring organizations leverage Microsoft's advanced security capabilities in combination with industry standards and best practices. The MCRA will be discussed later in the book.

Figure 1.4 presents the four key stages of SAF—assess, plan, implement, and manage:

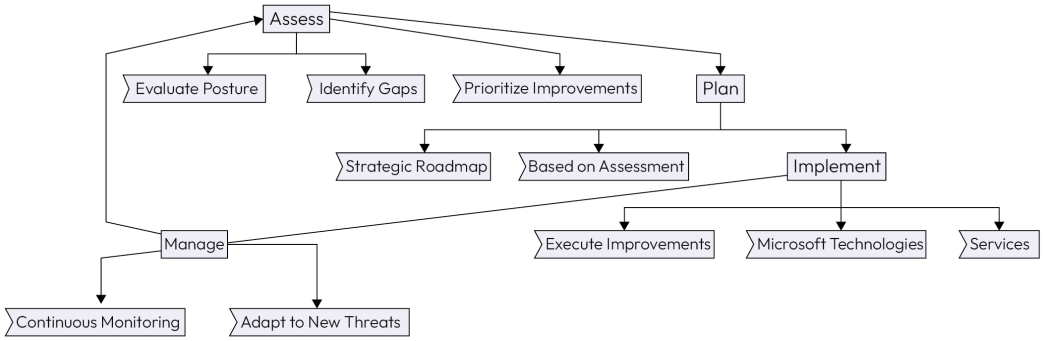


Figure 1.1: Microsoft Security Adoption Framework (SAF)

Let's break down the key stages of SAF:

1. **Assess**: Begin by evaluating your organization's current security posture. Identify any gaps, risks, and priority areas that need improvement. This step is crucial for gaining a clear picture of where you stand and what needs to be addressed. You will find some suggested tools later in this book!
2. **Plan**: Based on your assessment findings, develop a strategic roadmap to enhance your security posture. This plan should outline the actions you'll take, prioritize them based on risk, and leverage best practices for maximum impact.
3. **Implement**: Put your plan into action by applying the identified security improvements. This is where you leverage Microsoft's advanced security technologies and services to fortify your defenses effectively.
4. **Manage**: Security isn't a one-time fix; it's an ongoing process. Continuously monitor and manage your security environment, adjusting as new threats emerge and challenges arise. Adaptation is key to staying resilient against evolving threats.

All in all, SAF provides a structured yet flexible path for organizations to align with industry best practices while prioritizing actions based on their specific risks. It helps ensure that your security approach is comprehensive, scalable, and aligned with the unique needs of your organization.



Microsoft Digital Defense Report 2025

The Microsoft Digital Defense Report 2025 highlights a threat landscape where cybercrime has fully industrialized, with over 50 percent of attacks with known motives financially driven, primarily through ransomware and extortion, while classic espionage represents only a small fraction of activity. Microsoft now processes more than 100 trillion security signals per day, blocks around 4.5 million new malware samples daily, analyses tens of millions of identity risk detections, and inspects billions of emails every day for phishing and malware. Identity is firmly established as the primary attack surface, with adversaries focusing on credential theft, cloud access abuse, and misconfigured identities rather than traditional perimeter exploits. The report also shows how AI is accelerating both sides of the fight, enabling more convincing phishing and faster attack automation, while defenders increasingly rely on AI driven detection and response. Critical sectors such as healthcare and government remain high value targets, reinforcing the core message of the report: cyber risk is business risk, and effective defence now depends on strong identity protection, phishing resistant MFA, Zero Trust principles, and a clear focus on resilience and recovery rather than prevention alone.

For further details, you can explore the full Microsoft Digital Defense Report 2025 here: `https://www.microsoft.com/en-us/corporate-responsibility/cybersecurity/microsoft-digital-defense-report-2025/?msockid=27d50461de106ff936e41279dfab6e1d`.

The CIA triad: A cornerstone of cybersecurity

The **CIA triad** represents the key goals that every security professional strives to achieve when implementing security measures. It entails the principles of confidentiality, integrity, and availability. These principles form the foundation for developing and implementing effective cybersecurity strategies within an organization.

The CIA triad aligns closely with the MCRA and the NIST **Cybersecurity Framework (CSF)**. For example, the MCRA provides guidance on how to apply the CIA principles within a comprehensive architecture, ensuring that confidentiality, integrity, and availability are integrated into every layer of the technology stack, from cloud services to on-premises infrastructure. Cybersecurity experts use the CIA triad to address specific threats while also building a strong security culture within the organization.

Let's take a closer look at the CIA triad principles and how they fit into Microsoft's cybersecurity solutions:

- **Confidentiality (keeping your data private)**: Confidentiality is all about making sure sensitive information stays out of the wrong hands. Think of it as locking up your valuables. Only people with the key should have access. Today, where data is highly valuable, protecting personal, financial, and proprietary information is crucial.

  Microsoft's tools, such as Microsoft Purview and the Azure Information Protection scanner, are designed to help you do just that. They let you classify, label, and secure your data based on its sensitivity level. So, whether it's financial records or private client details, you can be confident that only the right people can access the information.

- **Integrity (keeping your data accurate and trustworthy)**: Integrity is all about making sure your data isn't tampered with or accidentally modified. Organizations rely on accurate and consistent information for important decisions, compliance, and earning trust.

  To help protect data integrity, Microsoft Sentinel and Microsoft Defender XDR leverage advanced analytics, ML, and threat intelligence to spot unusual activities that could compromise your data's accuracy. These solutions are designed to detect anomalies, such as unexpected changes in user behavior or system activity, which may indicate potential threats or malicious actions. By identifying these anomalies early on, security teams can intervene and stop bad actors before they cause harm, ensuring that your data remains reliable and unaltered.

- **Availability (making sure your data is there when you need it)**: In a world where businesses run 24/7, any downtime can cost you. Availability ensures that your information and systems are accessible whenever needed by those who have permission.

  Microsoft's Azure DDoS Protection and Azure Site Recovery help keep things running smoothly. Azure DDoS Protection defends against attacks that try to overwhelm your systems, while Azure Site Recovery enables quick recovery from disasters, making sure you stay up and running even during unexpected events.



Lessons from the field: balancing the CIA triad

While each component of the CIA triad is essential, it's important to recognize that there may be trade-offs between them. For example, implementing strict access controls to maintain confidentiality may impact the availability of information for authorized users. Similarly, ensuring the availability of systems may require less stringent integrity checks to reduce latency.

Applying the CIA triad to your security strategy

Figure 1.2 shows how to effectively apply the CIA triad to your organization's security strategy:

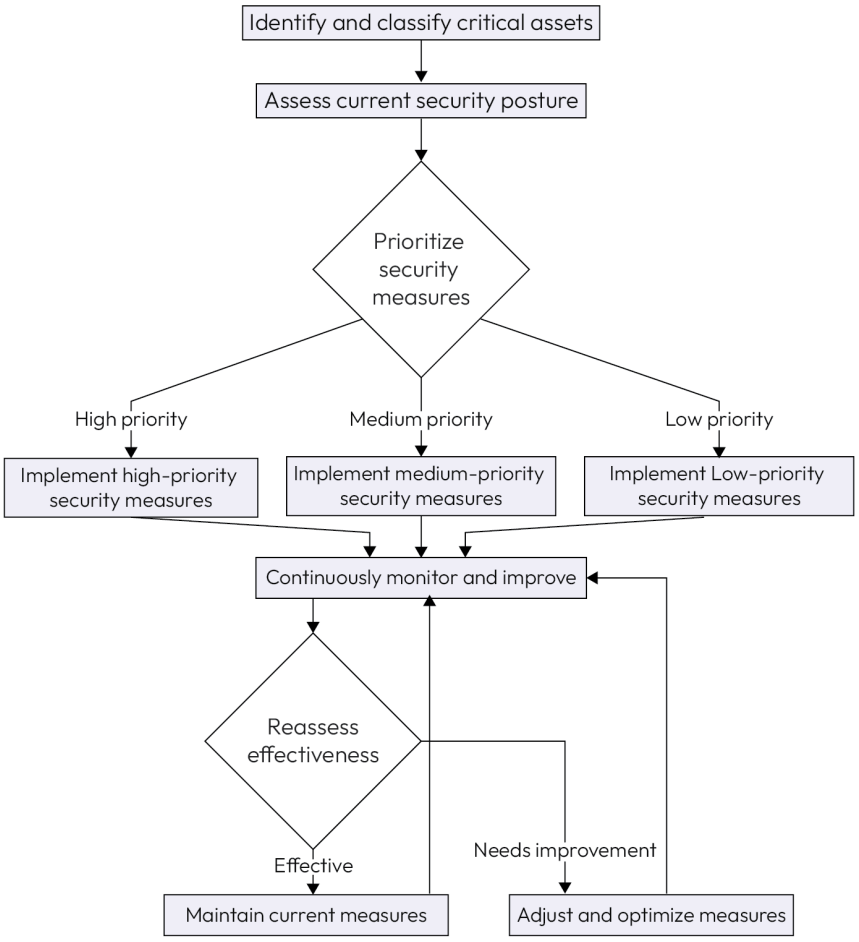


Figure 1.2: Application of the CIA triad in your organization

Let's now go through each of the key steps of the security strategy:

- **Identify and classify your critical assets**: Start by pinpointing the systems, applications, and data that are essential to your operations. Not all assets are equal, so categorizing them based on their sensitivity and criticality helps prioritize security efforts. For instance, customer data and financial records might need stronger protection compared to less sensitive information.

  For example, critical assets may include your CRM system containing sensitive customer **Personally Identifiable Information (PII)**, privileged administrative accounts with elevated access, or the domain controllers that underpin your entire identity platform. It's important to consider business-critical assets as well, not just the traditional "IT admin" components. These may include key operational applications, financial processing systems, customer-facing services, or any platform whose compromise would significantly disrupt the organization's ability to function or meet regulatory obligations.

- **Assess your current security posture**: Take a close look at your existing security controls and processes. Measure them against the principles of the CIA triad to identify gaps and weaknesses in your defenses. This will help you understand where your vulnerabilities lie and which areas need immediate attention.

  An example of a control is MFA, strengthening confidentiality by ensuring that only authorized users can access sensitive systems or data, even if credentials are compromised. A process example is quarterly access reviews, supporting integrity and confidentiality by validating that users retain only the permissions they genuinely require, reducing the risk of privilege creep or unauthorized access.

- **Prioritize and implement security measures**: Based on the results of your assessment, take a risk-based approach to prioritize the security measures that address the most significant risks. Focus first on controls that will have the most impact on improving the confidentiality, integrity, and availability of your critical assets.

  For example, if your assessment reveals that privileged accounts lack MFA, this becomes a high-priority action. Enforcing MFA for all administrative roles delivers immediate impact by significantly reducing the likelihood of account takeover, protecting both the confidentiality of sensitive data and the integrity of your identity platform.

Parallelly, you might identify that backups are occurring but have never been tested. Introducing a scheduled backup-restore testing process ensures that critical systems can be successfully recovered, directly strengthening availability. By addressing these high-impact risks first, you quickly improve overall resilience while laying a strong foundation for future enhancements.

- **Continuous monitoring and improvement**: Regularly monitor the effectiveness of your security measures, keep track of new threats, and adapt as needed. Continuous improvement ensures that your defenses stay aligned with the evolving threat landscape and your organization's changing needs.

  If you deploy endpoint protection across your device estate, pair it with **ongoing threat detection monitoring** through your SIEM or XDR platform. This ensures you're not just collecting telemetry but actively analyzing it for emerging attack patterns, misconfigurations, or signs of compromise.

  Alongside this, adopt a **regular security review cycle**, where you evaluate alerts, incident trends, and any newly discovered vulnerabilities. This process allows you to refine policies, update controls, and adjust your defensive posture as your organization grows or as the threat landscape shifts.

This approach will help you build a resilient security program, keeping your organization protected while maintaining compliance with frameworks such as the NIST CSF or MCRA.

Imagine your organization has moved its customer data and financial systems to the cloud. While this transition provides flexibility and scalability, it also introduces new risks. Hackers now have more ways to target your organization. That's where the CIA triad comes in, providing a practical framework to guide your security efforts.

For example, if your company uses Microsoft Defender XDR to monitor cloud activity, you're ensuring confidentiality by enforcing that only authorized users can access sensitive data. If a hacker tries to tamper with financial records, integrity is maintained through advanced threat detection that identifies and stops unauthorized changes. Meanwhile, tools such as Azure Site Recovery ensure availability, keeping your critical systems accessible even during a cyberattack.

The combination of theory and practical tools, such as Microsoft's cloud security solutions, makes the CIA triad an everyday guide for building a resilient and healthy secure ecosystem. We strongly recommend checking out the Microsoft security fundamentals documentation found here: `https://learn.microsoft.com/en-us/azure/security/fundamentals/overview`.

Now that we've covered strategy and frameworks, let's move on to assessments. This will help you identify any gaps and prioritize the most significant risks to your organization.

Assessing your current security posture and the need for modern DevSecOps

This section will guide you through the process of conducting comprehensive security assessment, prioritizing risks and vulnerabilities, and creating a plan to align your organization with modern DevSecOps practices. It's essential to keep the future state of security driven by DevSecOps principles in mind to ensure that your design and architecture not only address today's challenges but also remain scalable and resilient for the years ahead.

Conducting a security assessment and gap analysis

Let's start by exploring what a security assessment entails and its key components.

Security assessment

The first step in enhancing your security posture is to conduct a comprehensive assessment of your current security state. The assessment should encompass all aspects of the organization's digital ecosystem, including applications, data, processes, and overall infrastructure. Understanding your existing security controls, policies, and procedures will help you identify areas of strength and weakness and determine where gaps exist in your defenses.

There's a lot to consider when conducting a comprehensive security assessment. *Figure 1.3* shows the key areas:

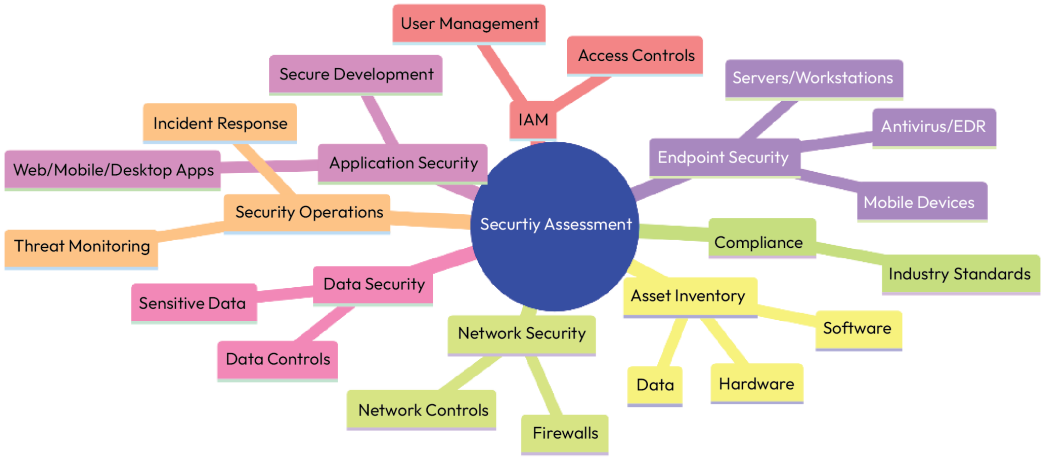


Figure 1.3: Key areas of a comprehensive security assessment

Let's take a deeper look into the key areas:

- **Asset inventory**: Identify and catalog all hardware, software, and data assets within your organization, including on-premises, cloud-based, and mobile resources. Keeping a detailed inventory helps ensure nothing is overlooked.
- **Network security**: Assess the security of your network infrastructure by focusing on firewalls, routers, switches, and **Wireless Access Points** (**WAPs**). Evaluate network segmentation, access controls, and monitoring capabilities to protect against unauthorized access and threats.
- **Endpoint security**: Examine the security of your endpoints, including servers, workstations, and mobile devices. Assess the effectiveness of your antivirus and anti-malware solutions, patch management processes, and EDR capabilities to safeguard against vulnerabilities.
- **Application security**: Evaluate the security of your applications, including web, mobile, and desktop platforms. Review your secure development practices, application testing procedures, and vulnerability management processes to ensure your applications are robust and secure.
- **Data security**: Identify and protect your sensitive data, such as customer information, intellectual property, and financial records. Assess the effectiveness of your data classification, encryption, and access control measures to prevent data breaches and unauthorized access.
- **Identity and Access Management (IAM)**: Evaluate the security of your IAM processes, including user provisioning, authentication, and authorization. Check the effectiveness of your password policies, MFA implementation, and **Privileged Access Management** (**PAM**) controls to ensure only authorized users have access to critical systems.
- **SecOps**: Examine the effectiveness of your SecOps, including incident response, threat detection, and security monitoring. Conduct maturity assessments of your SIEM and **Security orchestration, Automation, and Response** (**SOAR**) capabilities to enhance your ability to respond to threats promptly.
- **Compliance**: Evaluate your organization's compliance with industry standards and regulations, such as GDPR, HIPAA, PCI-DSS, and NIST. Ensuring compliance helps you avoid legal penalties and reinforces your commitment to security best practices.

A thorough assessment across these key areas will help you develop a clear view of your security posture, identify gaps, and give you the necessary ammunition to build your security roadmap and start implementing the necessary changes.

Lessons from the field

During our consulting engagements, we've found the following 10 questions invaluable for assessing an organization's current security posture and overall situation:

What are your organization's most critical assets, and how are they currently protected?

Identify and prioritize the key assets that are vital to your operations and understand the existing protection measures in place.

Have you conducted both internal and external risk assessments to identify potential vulnerabilities and threats to your systems and data?

Evaluate whether comprehensive risk assessments have been performed to uncover weaknesses and potential threats from all angles.

Do you have a comprehensive, documented incident response plan in place? How often is it reviewed and updated?



Ensure that there's a clear plan for responding to security incidents and that it's regularly updated to stay effective.

How do you ensure that your employees are well trained in cybersecurity best practices, and how frequently do you conduct security awareness training?

Assess the effectiveness and regularity of training programs aimed at educating employees about cybersecurity.

What measures have you implemented to protect against insider threats? Do you have concerns about shadow IT or data exfiltration?

Examine the strategies in place to prevent and detect threats originating from within the organization, including unauthorized IT usage.

How do you manage and secure remote access to your organization's network and resources, especially with the rise in remote work?

Review the controls and protocols established to safeguard remote connections and ensure secure access to organizational resources.



What is your patch management process for applications and devices to address known vulnerabilities?

Understand the procedures for keeping software and hardware up to date to mitigate potential security risks.

Do you have a SIEM solution in place? If not, how do you monitor all your security systems, and do you find your current approach effective?

Determine whether an SIEM system is utilized for comprehensive monitoring and how security events are tracked and managed.

Have you implemented a data backup and recovery strategy to ensure business continuity in the event of a security incident?

Verify that there are robust backup and recovery plans to maintain operations and data integrity during and after incidents.

How do you ensure compliance with relevant industry standards and regulations, such as GDPR, HIPAA, PCI-DSS, or others specific to your industry or jurisdiction?

Assess the processes in place to maintain adherence to necessary legal and regulatory requirements, avoiding potential penalties and enhancing trust.

Now, let's explore what a gap analysis involves.

Gap analysis and benchmarking

Once you've completed your security assessment, the next step is to conduct a gap analysis. This process helps identify where your organization's security posture may fall short of industry best practices and standards. Essentially, a gap analysis involves comparing your current security state with your desired future state, which can be guided by industry requirements or frameworks.

To streamline this process, consider leveraging the following established security frameworks and benchmarks:

- **NIST CSF**: A comprehensive framework designed for managing cybersecurity risk, offering detailed guidance on protection, detection, response, and recovery from cyber threats. It's widely recognized and utilized across various industries.

- **CIS Controls**: A set of best practices developed by CIS that provide actionable measures to achieve defense in depth, protecting your business against common attack vectors. These controls are regularly updated to reflect the latest threat landscape.
- **ISO/IEC 27001**: An international standard essential to the IT industry, ISO 27001 provides a systematic approach to managing sensitive company information, closely aligning with the CIA triad.
- **OWASP Top 10**: A widely recognized standard for developers and web application security, the OWASP Top 10 outlines the most critical security risks facing web applications, based on broad consensus from industry experts.



Visit these links to see the latest updates on these standards:

- NIST **Cybersecurity Framework (CSF)**: `https://www.nist.gov/cyberframework`
- CIS Controls: `https://www.cisecurity.org/controls/`
- ISO/IEC 27001: `https://www.iso.org/standard/27001`
- OWASP Top 10: `https://owasp.org/www-project-top-ten/`

Now that you've assessed your organization and identified the security gaps, what comes next?

Identifying and prioritizing security risks and vulnerabilities

Once you have mapped out your current security posture and the gaps within your resources, the next step is to identify and prioritize the specific risks and vulnerabilities that pose the greatest threat to your organization. To do so, you must assess the likelihood and potential impact of each risk to determine which vulnerabilities require your attention first.

Risk assessment methodology

To effectively identify and prioritize security risks, adopt a structured risk assessment methodology. Let's consider an instance where a mid-sized financial institution wants to improve its cybersecurity posture.

Here are a few widely recognized approaches that they leverage:

- **NIST SP 800-30**: This publication offers comprehensive guidance for conducting risk assessments within organizations. It provides a systematic approach for identifying, estimating, and prioritizing risks, ensuring that you have a solid foundation for your security strategy.

The organization performs a risk assessment by identifying critical systems, such as their online banking platform. They conduct interviews and workshops to identify threats and vulnerabilities, assess risks, and prioritize them. High-priority risks are documented, and a mitigation plan is developed, focusing on enhancing authentication methods and employee training.

- **ISO 27005**: This standard outlines a robust framework for security risk management, offering guidelines for identifying, analyzing, evaluating, and treating information security risks. It's particularly useful for organizations looking to align their risk management practices with international standards.

  Using this standard, the financial institution establishes a risk management framework and conducts a thorough risk assessment. They engage various departments to identify information assets, analyze risks, and evaluate them against regulatory compliance standards. A treatment plan is created to address identified gaps, such as implementing stronger encryption for sensitive data.

- **Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)**: Developed by Carnegie Mellon University's Software Engineering Institute, OCTAVE provides a set of tools, techniques, and methods for assessing and managing information security risks. This methodology emphasizes self-direction and organizational awareness, helping you build a resilient security posture.

  Using OCTAVE, a cross-functional team identifies their key assets, including customer data and transaction systems. They conduct a thorough assessment of potential threats and vulnerabilities, focusing on both technical and organizational factors. Strategies are developed to manage risks, including enhancing their incident response plan and improving collaboration across departments to ensure all staff are aware of security protocols.

Let's shift our focus to strategy and the importance of vulnerability management.

Vulnerability management

Vulnerability management is essential for organizations to effectively identify, assess, and address security weaknesses in their systems and applications. UK's **National Cyber Security Centre (NCSC)** emphasizes that a robust vulnerability management program involves a continuous cycle of activities, starting with maintaining an up-to-date asset inventory and performing regular vulnerability assessments.

Prioritizing vulnerabilities based on their potential impact is crucial for resource allocation and remediation efforts. By employing a risk-based approach and utilizing established frameworks and tools, organizations can not only mitigate risks but also strengthen their overall cybersecurity posture. This proactive management fosters a culture of continuous threats while safeguarding critical assets and ensuring compliance with industry standards.



You can download the NCSC Cybersecurity Toolkit for Boards, which provides a comprehensive approach to cybersecurity. It serves as an excellent starting point for introducing conversations about this critical subject within your business. You can find the file here: `https://www.ncsc.gov.uk/files/NCSC_Cyber-Security-Board-Toolkit.pdf`.

Now, let's discuss vulnerability management using Microsoft terminology and introduce the products available that you should leverage if they are part of your current license or if you plan to adopt the Microsoft security suite.

Building a strong vulnerability management program with Microsoft solutions

Regular vulnerability scanning is essential for a robust vulnerability management program. MDE provides advanced capabilities that help identify weaknesses in your systems and applications, ensuring that you maintain a proactive security posture.

The following are the key features of MDE's vulnerability scanning:

- **Automated scanning**: MDE continuously monitors your endpoints for known vulnerabilities, offering real-time insights into your security landscape
- **Comprehensive coverage**: The platform scans various operating systems, including Windows, macOS, Linux, and mobile devices, ensuring that all your assets are protected
- **Seamless integration**: MDE works in tandem with **Microsoft Defender Vulnerability Management**, giving you a centralized view of your organization's vulnerability landscape and enabling effective prioritization and tracking of remediation efforts
- **Threat intelligence**: Leveraging Microsoft's extensive threat intelligence database, MDE provides contextual information about vulnerabilities, helping you understand their severity, exploitability, and potential impact

Vulnerability assessment with Microsoft Defender Vulnerability Management

Once vulnerabilities are identified, the next step is to assess their severity and potential business impact. Microsoft Defender Vulnerability Management is a cloud-based solution designed to analyze scan results and highlight vulnerabilities that require immediate attention.

Key features of Microsoft Defender Vulnerability Management include the following:

- **Risk-based prioritization**: Utilizing ML and advanced analytics, this platform helps prioritize vulnerabilities based on their likelihood of exploitation, potential impact, and relevance to your specific environment.
- **Remediation guidance**: The platform offers actionable remediation steps for each identified vulnerability, including detailed instructions and links to relevant patches or updates.
- **Integration with Configuration Manager**: Seamless integration with **Microsoft Endpoint Configuration Manager** allows for easy deployment of patches and updates directly from the assessment interface.
- **Comprehensive reporting**: With pre-built reports and customizable dashboards, you can track your organization's vulnerability management progress and effectively communicate risks to stakeholders.

The real value of Microsoft Defender Vulnerability Management lies in its ability to turn complex vulnerability data into clear, actionable priorities. Instead of drowning in thousands of findings, you gain a focused view of the issues that genuinely threaten your business, and the steps needed to fix them. This means faster remediation, reduced attack surface, and a repeatable, measurable way to keep pace with evolving threats while supporting day-to-day operational demands.

Prioritizing vulnerabilities with Microsoft Secure Score

In addition to the capabilities of Microsoft Defender Vulnerability Management, you can leverage **Microsoft Secure Score** to further prioritize vulnerabilities and security enhancements. Secure Score provides a numerical representation of your organization's overall security posture, helping you understand your current standing.

The following are the key features of Microsoft Secure Score:

- **Actionable recommendations**: Secure Score provides guidance for improving your security posture, detailing steps to remediate vulnerabilities and implement best practices.

- **Comparative benchmarking**: You can compare your Secure Score against industry averages and peers, offering valuable insights into your security posture relative to others.
- **Integration with Microsoft solutions**: Secure Score integrates with a wide range of Microsoft security solutions, providing a comprehensive view of your security posture across your entire Microsoft environment.
- **Continuous monitoring**: The score updates in real time as you implement recommended improvements, allowing for ongoing visibility into your security posture.



Lessons from the field

Secure Score is an excellent tool for facilitating customer conversations, providing quick wins to help organizations embark on their security journey. It reflects how well security best practices are implemented to guard against potential threats.

Establishing a continuous vulnerability management program

To effectively safeguard your organization, it's essential to implement a continuous vulnerability management process that is regularly reviewed, at least once a year or whenever significant changes occur that may influence your security measures. This process should consist of four primary elements:

- **Identification**: Organizations must create an inventory of their digital assets, which includes proprietary software, third-party applications, sensitive information, and open-source components. Understanding the vulnerabilities associated with these assets is critical. Utilizing various assessment tools and scanning technologies can facilitate this identification process, which should occur frequently, potentially from weekly to daily, based on the organization's risk profile and IT environment complexity.
- **Evaluation**: After identifying vulnerabilities, they should be assessed and prioritized. Key metrics for this assessment might include the **Common Vulnerability Scoring System (CVSS)**, the likelihood of exploitation, resolution difficulty, potential financial repercussions, and any applicable regulatory standards.
- **Remediation**: Organizations need to address the vulnerabilities according to their prioritization. This can involve applying patches, leveraging automated updates from software vendors, utilizing patch management solutions, and implementing manual fixes where necessary.

- **Reporting**: Documenting identified vulnerabilities, the outcomes of evaluations, and remediation progress is crucial. Effective reporting not only aids in future remediation efforts but also simplifies discussions with executives and supports compliance requirements.

In practice, a continuous vulnerability management program is what keeps organizations from being blindsided by preventable breaches. By maintaining an accurate asset inventory, regularly evaluating risk, and enforcing timely remediation, you create a living system that adapts as your environment evolves. This approach not only reduces the window of exposure to known threats but also strengthens operational resilience. For leadership and technical teams alike, it provides a clear, repeatable framework that transforms vulnerability management from a periodic activity into a strategic capability.

Creating an effective remediation process

Once a vulnerability management framework is established, it's vital to develop a remediation process that clearly outlines how your organization will respond to vulnerabilities. This process should be documented and reviewed regularly, preferably every month or even more frequently. The remediation strategy should incorporate a variety of tools to tackle vulnerabilities, including automated patches and risk-based vulnerability management software to help prioritize potential threats. Incorporating advanced analytics and predictive technologies can also be beneficial in addressing risks proactively.

Automated patch management for operating systems and applications

Operating systems are foundational to IT infrastructure, making it essential to apply critical updates promptly. Automated patch management is important because it prevents delays and human error, which attackers rely on. In real environments where IT teams juggle competing priorities and large, distributed estates, manual patching simply cannot keep pace with the volume of vulnerabilities disclosed each month. Automation ensures that critical updates are deployed quickly and consistently, shrinking the attack surface and reducing the likelihood of exploitation. It also frees up valuable engineering time, allowing teams to focus on higher-value security work rather than routine maintenance. For most organizations, automated patching becomes one of the most reliable and measurable ways to reduce risk at scale.

An effective patch management framework includes the following features:

- **Information gathering**: Regularly scanning devices to identify those that need updates enables quicker deployment of patches. Some automated solutions can also provide insights into hardware and user details, offering a comprehensive view of endpoint status.

- **Patch download**: While obtaining patches is straightforward, complexities arise when many devices require different updates or when various operating systems are involved. Automated patch management solutions should efficiently manage these scenarios.
- **Package creation**: The system should be capable of assembling all the components required to apply a patch, ensuring smooth updates.
- **Patch distribution**: To minimize user disruption, patch management software should be able to schedule updates to run in the background during non-peak hours.

Following the application of patches, organizations should track which devices have been updated and which patches were applied, generating automated reports to inform IT teams of the next steps.

Conducting regular vulnerability scans

Organizations are advised to perform vulnerability scans of their IT assets regularly, using automated tools that comply with the **Security Content Automation Protocol (SCAP)**, **Cyber Assessment Framework (CAF)**, and NIST CSF.

The following are the various scan types that should be employed:

- **Network-based scans**: Identify vulnerabilities within wired and wireless networks by detecting unauthorized devices and ensuring secure connections with business partners
- **Host-based scans**: Evaluate endpoints such as servers and workstations and review configurations and patch histories to uncover weaknesses
- **Application scans**: Ensure that software applications are configured correctly and up to date
- **Wireless scans**: Detect rogue access points and verify proper configuration
- **Database scans**: Assess the security of databases

Both authenticated and unauthenticated scans play essential roles in understanding your true security posture. Authenticated scans allow the scanner to log in with valid credentials, providing deep visibility into configuration issues, missing patches, insecure settings, and vulnerabilities that only an internal or compromised user would be able to exploit. In contrast, unauthenticated scans simulate the perspective of an external attacker with no trusted access, helping to identify exposed services, misconfigurations, and entry points that could be targeted during an initial compromise. Together, these two scanning approaches deliver a comprehensive view, highlighting not only what an outsider can see but also what damage could occur if an account is compromised or an insider misuses their access.

Monitoring externally exposed assets

Organizations should pay special attention to vulnerabilities in assets exposed to external users, such as through the internet, because these systems represent the most accessible and attractive targets for attackers. Unlike internal services, which may require multiple steps of compromise to reach, internet-facing applications, APIs, VPN gateways, and email portals are continuously probed by automated scanners, threat actors, and opportunistic attackers looking for weaknesses. Even a single unpatched flaw can be exploited within hours of disclosure. Conducting monthly scans of these assets is recommended. Commonly overlooked external assets include devices, sensitive data, security codes, IoT sensors, remote operating equipment, client information, and remote work routers.

Remediating vulnerabilities promptly

Following the identification of vulnerabilities, organizations should adhere to their established remediation processes to ensure timely resolution, ideally performing remediation actions at least monthly.

While strong patching routines and solid BAU operational practices are essential, they are only part of the picture. These activities keep the organization safe day to day, but without a clear strategic goal, they risk becoming isolated tasks rather than contributing to meaningful long-term improvement. To truly elevate your security posture, every operational action, no matter how small, must align with the broader vision.

Developing a strategic roadmap for security enhancements with the Microsoft security stack

When it comes to enhancing your organization's security posture, a strategic roadmap is essential because it provides structure, direction, and measurable milestones. Without a defined plan, security efforts often become reactive, fragmented, or influenced by the crisis of the day rather than long-term resilience. A roadmap ensures that improvements are prioritized based on risk, aligned with business objectives, and delivered in a logical sequence that builds capability over time. With Microsoft's comprehensive security suite, you can build a robust framework that addresses various facets of security. Let's explore how each component plays a vital role in safeguarding your business assets.

Identity management: The first line of defense

Starting with identity, access control is crucial for protecting your organization's assets. Microsoft's identity management technologies, such as **Microsoft Entra ID**, **Microsoft Entra ID Protection**, **workload identities**, **Microsoft Defender for Identity**, and **Entra ID Governance**, ensure that only authorized users and devices gain access to resources. This suite of tools provides a strong foundation for maintaining a high level of security, allowing your team to focus on innovation and growth without compromising safety.

Embracing Zero Trust architecture

At the core of modern cybersecurity is the **Zero Trust architecture**. Microsoft equips organizations with powerful tools such as **Microsoft Entra Conditional Access**, **Entra Public Access**, and **Entra Private Access**, combined with **Microsoft Defender for Cloud Apps**. These technologies help you enforce centralized policy controls that adapt to various factors, including user identity, device health, network location, and risk profile. This granular approach to access management ensures that security measures are not only effective but also flexible enough to meet the demands of a dynamic business environment.

Securing user endpoints

User endpoints are critical in the access control landscape, as compromised devices can provide attackers with a gateway to your organization's assets. Microsoft's endpoint security solutions, such as **Microsoft Intune**, **MDE**, **Microsoft Defender Application Guard**, **Azure Virtual Desktop**, and **Windows 365**, are designed to protect and manage user devices. These tools ensure that only compliant and trusted endpoints can access corporate resources, creating a fortified perimeter around your sensitive information.

Modern SecOps for a swift response

Modern SecOps is essential for promptly detecting, investigating, and responding to security incidents. Microsoft's offerings, including **Microsoft Defender XDR** and **Microsoft Sentinel**, which have now merged into the unified SecOps platform, provide a *true* single pane of glass to leverage analytics, automation, and AI to identify threats across multiple domains, streamline investigations, and enable rapid responses.

Protecting data assets

Data assets represent some of your organization's most valuable and sensitive resources, often subject to stringent regulations. An example is customer PII, which is regulated because it contains data that can directly identify an individual; misuse or exposure can cause harm, identity theft, or financial loss. Laws such as GDPR mandate strict controls to protect individuals' privacy and rights.

Financial and payment systems are also subject to regulation because they process monetary transactions, credit card data, and financial records. Compromise could lead to fraud, market manipulation, or systemic financial risk. Frameworks such as PCI-DSS and financial conduct regulations enforce strong safeguards.

Health, employee, and HR records are regulated due to the highly sensitive nature of medical, employment, and personnel information. Exposure could lead to discrimination, reputational harm, or legal liability. Data protection laws require heightened confidentiality and strict access controls.

Microsoft's data security solutions, such as **Defender for Office 365**, **Microsoft Purview**, **Microsoft Purview DLP**, **Microsoft Purview Information Protection**, **Microsoft Priva**, and **Defender for Cloud (SQL database/files)**, help organizations safeguard, govern, and manage their data across various platforms and services. These tools ensure that your data remains secure while allowing you to meet compliance requirements with confidence.

Securing applications against threats

Protecting **applications** is critical for maintaining the integrity of business operations and preventing unauthorized access. Microsoft's suite of technologies, including **GitHub Advanced Security (GHAS)**, **Defender for Cloud Apps**, and **Microsoft Defender for APIs**, offers comprehensive application security capabilities. These solutions facilitate secure application development, protect against threats targeting **Software as a Service (SaaS)** applications, and secure APIs throughout their life cycle, ensuring that your applications are robust and resilient.

Infrastructure security for hybrid environments

Lastly, securing your **infrastructure**, whether in the cloud or on-premises, is vital for a holistic security strategy. Microsoft provides tools such as **Entra Permissions Management**, **Microsoft Defender for Cloud**, **Azure Arc**, and **Azure Networking** to help organizations protect their infrastructure. These technologies manage permissions, monitor for threats, extend security controls across hybrid and multi-cloud environments, and offer advanced networking security features, enabling you to create a secure and compliant infrastructure.

Now that we've laid the groundwork for a strong security framework, let's shift our focus to DevOps and DevSecOps.

Enter DevOps and DevSecOps

In traditional software development, projects often move through distinct phases of planning, design, development, integration, and testing, each taking considerable time, sometimes extending for months or even years. While this method offers structure, it can be too slow for many organizations that are trying to keep pace with the relentless demand for product updates. Moreover, security often gets pushed to the end of the process, increasing the risk of cyberattacks.

To stay competitive and agile, many companies have embraced a **DevOps** approach, which prioritizes smaller, more frequent, and higher-quality code updates over larger, feature-heavy releases that require longer timelines. This model fosters collaboration between software development and operations teams, integrating continuous testing and integration throughout the project life cycle. By adopting automation and streamlined workflows, these teams can deliver updates swiftly without compromising quality.

Building on DevOps principles, **DevSecOps** introduces a security-first mindset at every stage of the **Software Development Life Cycle** (**SDLC**). This approach focuses on addressing security challenges right from the start, ensuring that security practices are embedded throughout the development process. It promotes shared accountability for quality, integration, and security across the entire team. Practically, this means having security discussions during the initial planning phases and conducting security assessments early on, rather than waiting until later stages. This proactive strategy, known as **shift-left security**, integrates security into the development workflow rather than treating it as an afterthought.

Integrating DevSecOps practices

As digital transformation accelerates and organizations increasingly adopt cloud-native solutions, weaving security throughout the SDLC is more crucial than ever. DevSecOps is a methodology designed to infuse security into the DevOps workflow. It makes security a continuous, shared responsibility across development, security, and operations teams. Security testing, code analysis, configuration validation, and compliance checks are automated and integrated into each stage of the pipeline. This ensures that risks are identified early, remediated quickly, and prevented from reaching production. The result is a culture and workflow where security becomes part of everyday engineering practices, enabling organizations to innovate at speed without sacrificing protection or reliability.

A typical IT operations team is responsible for managing servers, patching, and endpoint security, as well as keeping core services running. In a DevSecOps model, they integrate security into these everyday tasks rather than treating it as a separate activity.

Consider an example where when deploying a new virtual machine or container image, the IT operations team no longer relies on manual checks or waits for a security review after deployment. Instead, the build pipeline automatically validates the image for known vulnerabilities, misconfigurations, missing patches, or insecure baseline settings (such as open ports or outdated libraries).

If issues are found, the pipeline fails the deployment and provides clear remediation guidance. This prevents risky or non-compliant configurations from ever reaching production. IT operations can then fix the baseline template, rerun the deployment, and move forward confidently. By adopting DevSecOps approaches within your organization's SDLC, you can identify and address vulnerabilities earlier, reduce the likelihood of security breaches, and elevate the security standards of both your applications and infrastructure.

Key components of DevSecOps

DevSecOps revolves around several key principles that highlight the importance of collaboration, automation, and continuous improvement in the software development process:

- **Shift-left security**: This principle advocates for integrating security measures from the very beginning of the SDLC. By focusing on security early, vulnerabilities can be identified and remediated before they reach production, minimizing potential impacts and costs.
- **Continuous Integration and Continuous Delivery (CI/CD) with security gates**: DevSecOps promotes the use of automated pipelines for development, integration, and deployment. These pipelines should include security checkpoints to ensure that code changes comply with security and regulatory standards before being deployed.
- **Infrastructure as Code (IaC)**: Managing infrastructure through code using tools such as **Azure Resource Manager** (**ARM**) templates or Terraform allows for versioning, testing, and automation, reducing the risk of configuration errors and ensuring consistency across environments.
- **Collaboration and shared responsibility**: DevSecOps emphasizes the need for cooperation and mutual accountability among development, security, and operations teams. By breaking down departmental silos, organizations can enhance communication and cultivate a culture where security is a collective priority.

- **Continuous monitoring and feedback**: At the core of DevSecOps is the commitment to ongoing monitoring of software and infrastructure for security threats. This proactive stance allows teams to identify and address security issues swiftly, reducing the risk of exploitation.

Together, these principles form the foundation of DevSecOps. To make this model work in practice, teams need a platform capable of supporting secure pipelines, policy-driven automation, and collaborative workflows. This is where Azure DevOps becomes a powerful enabler. With integrated boards, repositories, pipelines, and security tooling, Azure DevOps provides the structure and automation required to operationalize DevSecOps on a scale.

DevSecOps and Microsoft tools and services

Microsoft offers a comprehensive suite of tools and services that facilitate the implementation of DevSecOps practices within your organization. These tools cover the entire SDLC, from development to deployment, monitoring, and response:

- **Azure DevOps**: This platform provides a robust set of tools for DevOps practices, including version control, CI/CD, automated testing, and team collaboration. It also integrates security features such as **Defender for Cloud** and the **Secure DevOps Kit for Azure** (**AzSK**) to help secure Azure deployments.
- **GHAS**: Designed to detect and remediate vulnerabilities in your code, this tool offers static analysis for scanning code and identifying security risks, along with mechanisms to detect sensitive information such as passwords and API keys.
- **Azure Policy**: With Azure Policy, you can establish and enforce rules across your Azure environments, ensuring compliance with security standards and regulatory requirements.
- **Azure Blueprints**: This tool allows for the deployment of standardized Azure resource sets, including policies and templates, to facilitate consistent configurations across Azure deployments.
- **Microsoft Defender for Cloud**: Offering protection for Azure and hybrid cloud workloads, Defender for Cloud includes features such as application control, file integrity monitoring, and network hardening, all tailored for a DevOps context.
- **Azure Monitor**: Providing detailed monitoring for Azure and on-premises environments, Azure Monitor includes log analytics and application insights to oversee application and infrastructure health, enabling quick detection and response to security threats.

With the core principles of DevSecOps established, the next step is translating them into practical, repeatable workflows. This is where the Microsoft ecosystem becomes a powerful accelerator. Each tool provides a specific capability designed to operationalize security across your development life cycle. Let's go through the steps that you should gradually begin to implement in your organization.

Embracing DevSecOps in your organization

Integrating DevSecOps effectively requires careful planning and a systematic approach. To make security an integral part of your development process, it's important to adopt practices that streamline team workflows while significantly enhancing security measures. Here's a step-by-step approach to seamlessly weave security into the regular DevOps cycle.

Phase 1: Planning and development

Incorporating security from the very beginning of the development cycle is essential for minimizing vulnerabilities and saving time by addressing potential issues before they reach the code integration stage.

During this phase, you should apply the following:

- **Threat modeling**: Utilize threat modeling to identify and mitigate potential threats, embedding security into the application from the outset. This proactive measure helps create a robust security foundation.
- **Automated security tools**: Leverage security plugins in IDEs to provide developers with instant feedback on security risks in their code. This allows for quick adjustments before the code is merged into the main repository.

  Table 1.1 presents a few examples of automated security tools:

Tool	Supported IDEs	Key Capabilities	Typical Use Case
GHAS extensions	Visual Studio Code	Secret scanning, dependency vulnerability alerts, CodeQL analysis, and insecure code pattern detection	Developers receive immediate warnings about exposed secrets, risky code, or vulnerable libraries directly while coding

Tool	Supported IDEs	Key Capabilities	Typical Use Case
SonarLint	Visual Studio, Visual Studio Code, IntelliJ, Eclipse	Real-time static code analysis, security bug detection, code quality insights, and inline remediation tips	Ensure developers spot security flaws (e.g., injection vectors) and technical debt as they write or refactor code
Snyk IDE integration	Visual Studio Code, IntelliJ, Eclipse	Vulnerable dependency detection, container/IaC scanning, fix recommendations, and version upgrade guidance	Help developers avoid shipping apps with vulnerable packages or insecure configuration files

Table 1.1: Examples of automated security tools

- **Engage security experts**: Involve security specialists in the code review process to gain valuable insights and expert advice on potential enhancements. Their expertise can significantly bolster the security posture of your applications.

By embedding security at the earliest stage of the development life cycle, phase 1 ensures that vulnerabilities are identified long before they become costly or complex to fix. This approach not only reduces risk but also accelerates later phases of development by ensuring that secure coding practices are built in from the very beginning.

Phase 2: Code commit

CI is a cornerstone of DevSecOps, enabling developers to merge changes into a central code repository multiple times a day. This practice helps identify integration issues early on. This marks a security and quality checkpoint in the DevSecOps workflow. The pipeline pulls the updated code/config, validates that it compiles correctly, and runs a series of automated tests to ensure nothing has broken because of the change. This could be run in a development environment.

In a mature DevSecOps environment, this is also where security begins. **Static Application Security Testing (SAST)**, dependency vulnerability checks, secret detection, and coding standard validations are executed automatically. These principles can be translated into SecOps config. Any issues, whether functional defects, insecure patterns, or vulnerabilities, are surfaced immediately to the author.

By performing these checks at every commit, teams identify integration conflicts and security risks early, when they are easiest and least expensive to fix. This creates fast feedback loops, reduces rework, and ensures that only high-quality, secure code moves forward in the pipeline.

These are the common checks done at this stage:

- **Automated security scans**: Integrate automated security measures such as static code analysis and secret detection to scan for vulnerabilities in third-party libraries and dependencies, alongside unit testing and static code analysis. This ensures that security is considered alongside functionality.
- **Strict access controls**: Implement robust access controls such as Entra ID RBAC with PIM and life cycle management to protect the CI and CD pipeline from unauthorized code execution or credential theft. This step is crucial for safeguarding your development environment.

By integrating automated security scans directly into the pipeline and enforcing strict access controls, organizations create a CI environment that is resilient, trustworthy, and resistant to tampering.

Phase 3: Building and testing

Executing security scripts in test environments is essential for uncovering issues that may have slipped through earlier checks. During this phase, consider **dynamic security testing**. Perform dynamic security testing, infrastructure scans, container assessments, cloud configuration checks, and security acceptance testing to identify potential vulnerabilities before deployment.

Phase 4: Production

Once your application is ready for deployment, some organizations conduct penetration testing in the live environment. This simulates an attacker's perspective to uncover exploitable weaknesses that could jeopardize security.

Phase 5: Operations

Recognize that even the most comprehensive DevSecOps strategy may not catch every risk. Ongoing monitoring of applications and infrastructure for new vulnerabilities is vital. Utilize analytics derived from previous tests, runs, and external threat intel feeds to assess security improvements and identify areas that need further refinement. This continuous feedback loop ensures that security remains a top priority even after deployment.



Keep in mind that adopting DevSecOps isn't a one-and-done deal; it's an ongoing journey. If you want to get started in your DevSecOps journey, we strongly recommend checking out this GitHub repository: `https://github.com/sottlmarek/DevSecOps`.

This library contains a list of tools and methodologies accompanied by resources. The main goal is to provide the engineers with a guide through open source DevSecOps tooling. This repository covers only cybersecurity in the cloud and the DevSecOps scope. Later in the book, we will look into a DevSecOps tool called Maester, which is a Microsoft Security test automation framework.

We will deploy Microsoft Sentinel via code later in *Chapter 8*.

The Microsoft Security competitive advantage and Security Copilot

Organizations of all sizes are feeling the pressure to safeguard themselves against increasingly sophisticated cyber threats. As a global tech leader, Microsoft is uniquely equipped to tackle this challenge, leveraging its vast resources, deep expertise, and extensive partnerships to innovate and foster collective defense.

Microsoft's security ecosystem is truly remarkable. With a dedicated team of over **10,000 cybersecurity professionals**, the company processes an astonishing **78 trillion signals every day**—that's more than **750 million signals each second**! This immense data stream allows Microsoft to harness advanced data analysis and AI technologies to swiftly detect and respond to potential threats.

The **Threat Intelligence teams** at Microsoft keep an eye on over **300 distinct threat groups** around the globe, including 160 tied to nation-state actors and several ransomware syndicates. This stance enables Microsoft to foresee and neutralize potential cyber threats before they escalate, providing critical insights that empower organizations worldwide to strengthen their defenses.

Microsoft's commitment to cybersecurity extends beyond its own operations. With a network of over **15,000 partners**, the company offers a diverse range of tailored security solutions that enhance defense capabilities for clients across various sectors. Moreover, Microsoft actively collaborates with the global security community, running initiatives such as bug bounty programs and security competitions to identify and fix vulnerabilities.

The effectiveness of Microsoft's security initiatives shines through in significant achievements: disrupting over **100,000 malicious domains**, blocking **4,000 identity threats every second** through its robust identity verification systems, and managing security for **135 million devices**. These efforts provide valuable insights into the ever-changing nature of cyber threats.



These statistics will probably be obsolete by the time this book is published, which shows how fast technology and Microsoft as a cybersecurity vendor are evolving!

The future of AI and Microsoft in cybersecurity

Microsoft is a leader in integrating AI into cybersecurity. As threats become more intricate, the company remains steadfast in its mission to create a secure digital environment. However, understanding the potential pitfalls of AI, Microsoft is committed to developing these technologies responsibly. The company has established ethical guidelines for AI development (`https://www.microsoft.com/en/ai/responsible-ai`), focusing on values such as fairness, reliability, privacy, security, inclusivity, transparency, and accountability. These principles are designed to guide Microsoft's AI solutions toward outcomes that benefit society.

In its commitment to responsible AI, Microsoft collaborates with industry leaders to establish standards and create technologies that enhance the reliability and transparency of digital content. A prime example of this is Microsoft's involvement in the **Coalition for Content Provenance and Authenticity** (**C2PA**), where it partners with companies such as Adobe and Intel. This initiative aims to develop an open standard for verifying the origins and authenticity of online content, which is crucial for combating misinformation and the spread of deepfakes, ultimately fostering trust in the digital space.



Organizations are increasingly concerned about the exposure of their data to AI models. Microsoft has made significant efforts to address these concerns by clarifying doubts and establishing a solid foundation for the data life cycle. In 2025, Microsoft released a Responsible AI Transparency Report, which we recommend reading. You can find the report here: `https://www.microsoft.com/en-us/corporate-responsibility/responsible-ai-transparency-report/?msockid=1b3ecb23b03b6f460b93de4ab1db6e42`.

Microsoft's leadership in applying AI to cybersecurity reflects a long-term commitment to strengthening global digital resilience. As threats grow more complex, the company continues to innovate while ensuring its AI capabilities are built on a foundation of responsibility and trust. By embedding principles such as fairness, reliability, transparency, privacy, and accountability into every stage of AI development, Microsoft aims to ensure that these technologies do not compromise societal well-being.

Summary

In this chapter, we introduced you to the world of Microsoft Security, establishing a foundation for understanding key concepts such as Microsoft Defender XDR, Microsoft Sentinel, and the innovative Security Copilot. We discussed the importance of XDR in today's cybersecurity landscape and how it integrates various security layers to provide comprehensive threat detection and response. Additionally, we explored the transformative role of AI in enhancing SecOps and the emergence of DevSecOps as a vital framework for embedding security throughout the SDLC. We also touched on established frameworks such as NIST CSF, CIS Controls, ISO/IEC 27001, and OWASP Top 10, which provide valuable guidelines for building effective security strategies.

As we transition to the next chapter, we will focus on aligning your organization's security strategies with the Microsoft Cloud Adoption Framework and Well-Architected Framework. You will also gain practical experience in assessing the security of your tenant using top-notch tools.

Get This Book's PDF Version and Exclusive Extras

UNLOCK NOW

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

2

Aligning with Cybersecurity Frameworks

Framework alignment turns Sentinel and Defender XDR from reactive security tools into a governed, measurable, and continuously improving SecOps capability. Every alert and playbook directly supports a recognized control objective. For instance, you can map Azure AD sign-in logs and risky sign-in analytics to NIST CSF Detect and CIS Control 8 (Audit Log Management) so you can prove to auditors that identity threats are being actively monitored and responded to.

This chapter explores essential cybersecurity frameworks and the best practices in cybersecurity capabilities and controls. It also covers the definition and implementation of the Microsoft Cloud Adoption Framework, which now includes guidance on AI governance, data compliance, and aligning cloud operations to business outcomes. The Microsoft Cloud Adoption Framework helps organizations build sustainable, scalable, and secure environments in increasingly complex digital landscapes. The focus of this chapter is to provide you with the knowledge to align your security strategies with Microsoft's advanced frameworks.

This chapter will cover the following topics:

- Essential cybersecurity frameworks
- The Secure Future Initiative (SFI) – a new foundation for trust
- Diving into the Microsoft CAF and Azure WAF
- Defining and implementing Zero Trust strategies for your infrastructure
- The evolution of cybersecurity – from physical locks to Zero Trust
- Exploring Zero Trust technologies and solutions

Let's start by exploring the most renowned cybersecurity frameworks.

Essential cybersecurity frameworks

Across the United States, the United Kingdom, and Europe, certain cybersecurity frameworks and standards have become fundamental for organizations in many industries. These frameworks provide structured approaches to evaluating, implementing, managing, and improving cybersecurity measures, helping businesses stay resilient in the face of evolving threats.

In this section, we'll explore the most prominent and widely adopted cybersecurity frameworks, highlighting their role in guiding organizations toward a stronger, more proactive security posture.

National Institute of Standards and Technology Framework

The **National Institute of Standards and Technology** (**NIST**) has developed this cybersecurity framework (CSF) as a robust, flexible framework to help organizations to bolster their cybersecurity posture and effectively manage risks in the digital landscape. It serves as a cornerstone of how organizations across the globe approach cybersecurity risk. This framework amalgamates a wide array of existing standards, guidelines, and best practices into a cohesive structure that can be adapted to suit the unique needs of businesses across various industries and sectors.



In February 2024, NIST released version 2.0 of its cybersecurity framework, introducing a sixth core function: Govern. This addition emphasizes the importance of organizational oversight, roles, and responsibilities in cybersecurity risk management. The update also refines existing categories to better align with modern cybersecurity practices.

The five pillars of the NIST CSF

At the heart of the NIST CSF are five core functions that provide a strategic view of the life cycle of managing cybersecurity risk. *Figure 2.1* presents these pillars:

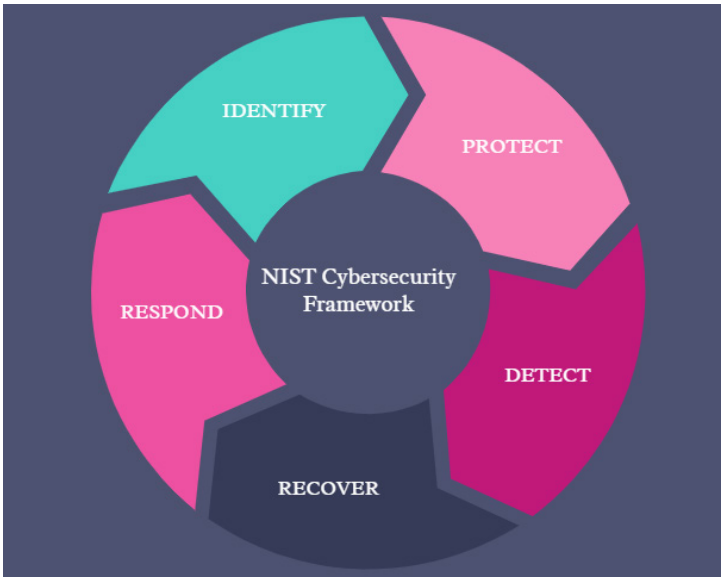


Figure 2.1: NIST Cybersecurity Framework CSF

Let's look at these pillars in detail:

- **Identify**: This foundational step is like taking a comprehensive inventory of your organization's digital landscape. It involves understanding your assets, systems, data, and the potential risks they face. By mapping out what's most critical, you can prioritize your cybersecurity efforts effectively.
- **Protect**: Once you've identified what needs safeguarding, the next step is to implement protective measures. This includes enforcing access controls, providing cybersecurity training to your team, encrypting sensitive data, and maintaining your systems and software. Think of it as building a fortress around your valuable assets.
- **Detect**: Despite strong defenses, breaches can still occur. The Detect function is about setting up mechanisms to quickly identify any anomalies or potential security events. Continuous monitoring and real-time alerts help ensure that threats are spotted and addressed promptly.
- **Respond**: When a cybersecurity incident is detected, how you react is crucial. This function involves having a well-defined response plan to contain the impact, communicate with stakeholders, analyze the incident, and implement improvements to prevent future occurrences.

- **Recover**: The final pillar focuses on bouncing back and restoring normal operations after an incident. It includes executing recovery plans, communicating transparently with stakeholders to rebuild trust, and updating your strategies based on lessons learned.

Now that we understand the core pillars, let's see how we can apply them.

Getting started with NIST CSF 2.0

To support organizations in adopting the CSF, NIST offers a wealth of resources:

- **NIST CSF Roadmap**: This guide provides a step-by-step approach to aligning your cybersecurity program with the framework
- **NIST CSF Self-Assessment Tool**: You can evaluate your current cybersecurity posture and identify areas for improvement with this handy tool



You can find the latest version of these tools and resources at `https://www.nist.gov/cyberframework`. Before diving into the frameworks, take some time to explore these materials. They offer valuable insights and practical guidance to set you on the right path.

NCSC Cyber Assessment Framework (CAF) – United Kingdom

The UK's **National Cybersecurity Centre** (**NCSC**) has developed the **Cyber Assessment Framework** (**CAF**) to help organizations assess their cybersecurity posture, manage risks effectively, and implement meaningful improvements. Whether you're a small business or a large enterprise, the CAF is designed to be flexible and adaptable to your unique needs.

In essence, the NCSC CAF focuses on four key objectives that come together to build a strong and resilient cybersecurity foundation. *Figure 2.2* is a mental map that illustrates how these objectives interconnect to guide organizations toward a more secure posture.

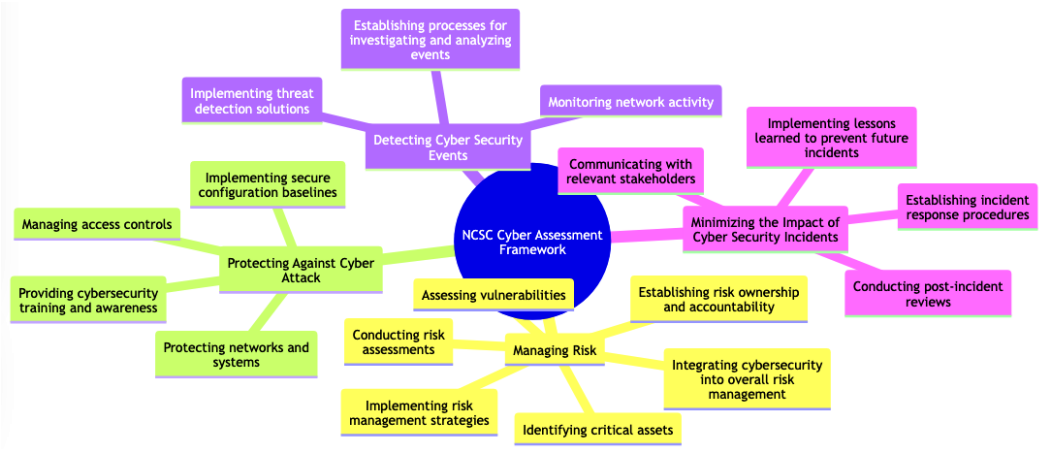


Figure 2.2: NCSC Cyber Assessment Framework CAF

The following are the key objectives of the NCSC CAF:

- **Managing Risk**: This objective is all about understanding and addressing the cybersecurity risks that an organization faces. It involves identifying critical assets, assessing vulnerabilities, and putting the right risk management strategies in place. Key activities include conducting risk assessments, establishing clear ownership and accountability for risks, and integrating cybersecurity into the organization's broader risk management framework.
- **Protecting Against Cyber Attack**: Once the risk landscape is understood, this objective focuses on implementing the necessary measures to protect the organization's assets and data. This can involve a wide range of activities, such as setting secure configuration baselines, managing access controls, securing networks and systems, and providing cybersecurity training and awareness programs for employees.
- **Detecting Cybersecurity Events**: Even with robust protections in place, it's essential for organizations to be able to quickly identify and respond to potential threats. The goal here is to establish mechanisms for promptly detecting security incidents. This includes monitoring network activity, deploying threat detection solutions, and setting up clear processes for investigating and analyzing any detected events.
- **Reducing the Impact of Cybersecurity Breaches**: When incidents do occur, being prepared to contain, mitigate, and recover is crucial. This objective emphasizes having a solid incident response plan in place. Key activities include establishing incident response procedures, communicating effectively with stakeholders, conducting thorough post-incident reviews, and implementing lessons learned to prevent similar incidents in the future.

Benefits of NCSC CAF

NCSC's CAF is praised for its accessibility and adaptability, making it suitable for organizations across various industries and sizes. It offers a flexible approach for assessing and enhancing cybersecurity practices without being overly rigid or prescriptive. The CAF focuses on outcomes rather than just ticking boxes, allowing organizations to align the framework with their specific goals and regulatory contexts. This approach supports a customized cybersecurity strategy that caters to unique risk landscapes.

For example, a housing association can use the NCSC CAF to assess how well it protects tenant data, map the outcomes to UK regulatory expectations, and then tailor improvements such as strengthening access controls or incident response based on its own systems, risks, and budget, rather than following a one-size-fits-all checklist.

Another key advantage is the CAF's compatibility with established standards, such as ISO/IEC 27001 and the NIST Cybersecurity Framework. This alignment helps organizations leverage existing investments in cybersecurity while incorporating the additional guidance provided by the CAF. For example, an organization that's already ISO 27001 certified and using the NIST CSF can overlay the CAF on top of its existing controls and map CAF outcomes to its ISO policies and NIST functions so it doesn't start from scratch, but instead reuses current risk assessments, logging standards, and incident processes while filling any UK-specific gaps highlighted by the CAF. Moreover, the framework's flexibility extends to both IT and OT, making it especially valuable for sectors that rely on critical infrastructure, such as utilities and healthcare.

Adoption and implementation of NCSC CAF

To support organizations in adopting and implementing the CAF, the NCSC provides a range of resources and tools. These include the CAF Guidance, which offers detailed explanations and practical advice on applying the framework, and the CAF Assessment Tool, which makes it possible for organizations to run self-assessments of their cybersecurity posture against the CAF objectives and identify areas for improvement.



Adopting the NCSC CAF helps break down organizational silos by promoting a shared responsibility for cybersecurity. By involving stakeholders from various departments and levels, you enhance communication and collaboration, ensuring everyone understands their role in protecting the organization. To access the latest version of NSCS CAF guidance, follow this link: `https://www.ncsc.gov.uk/collection/caf/introduction`.

Cyber Essentials and Cyber Essentials Plus – United Kingdom

Cyber Essentials and **Cyber Essentials Plus** are initiatives supported by the UK government, aimed at assisting organizations in defending against common cyber threats. These programs offer straightforward guidelines and measures that companies can adopt to strengthen their cybersecurity stance, especially as cyberattacks become more advanced and frequent.

Let's now examine what is covered by each scheme.

Cyber Essentials

Cyber Essentials is a self-assessment that focuses on five key control areas, as shown in *Figure 2.3*:

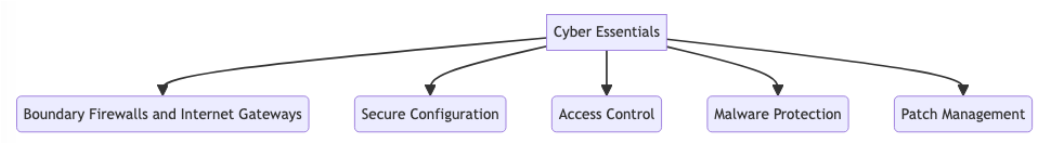


Figure 2.3: Cyber Essentials control areas

- **Boundary Firewalls and Internet Gateways**: Making sure that devices are secure from unauthorized access and that only pre-approved network services can be accessed from the internet
- **Secure Configuration**: Implementing appropriate security measures for all devices and software, including disabling unnecessary functionality, changing default passwords, and regularly updating systems
- **Access Control**: Limiting system and data access to those who are authorized and enforces stringent authentication methods such as strong passwords and MFA
- **Malware Protection**: Protecting against malware infection by implementing and regularly updating antivirus software, as well as promoting safe browsing practices among employees
- **Patch Management**: Ensuring that all devices and software are kept up to date with the most recent security patches to mitigate known vulnerabilities



Organizations can assess their compliance with these controls and, upon successful completion, receive a Cyber Essentials certification. The aim of this certification is to demonstrate the organization's commitment to cybersecurity and provide assurance to customers, partners, and stakeholders that the business has taken appropriate measures to protect against common cyber threats.

Cyber Essentials Plus

Cyber Essentials Plus is a more rigorous certification that builds upon the foundation of Cyber Essentials. In addition to the self-assessment, Cyber Essentials Plus involves an independent, third-party vulnerability assessment to verify the organization's implementation of the five key control areas.

The vulnerability assessment covers the areas illustrated in *Figure 2.4*:

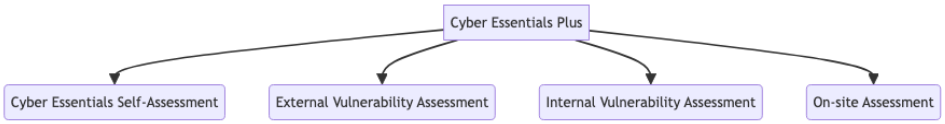


Figure 2.4: Cyber Essentials Plus control areas

Let's look at these areas in detail:

- **External Vulnerability Assessment**: An external scan of the organization's internet-facing systems to identify potential vulnerabilities and misconfigurations
- **Internal Vulnerability Assessment**: An assessment of the organization's internal network to identify potential weaknesses and ensure that the controls have been effectively implemented
- **On-site Assessment**: A visit by a certified assessor to review the organization's cybersecurity practices, policies, and procedures, as well as to conduct interviews with key personnel



Lessons from the field

For UK businesses aiming to secure cyber insurance or win contracts, particularly in sectors such as government or defense, the Cyber Essentials and Cyber Essentials Plus certifications are invaluable. These certifications establish a baseline of cybersecurity, ensuring that your organization meets essential standards to defend against the most common cyber threats. In fact, they're not just beneficial; they're fast becoming expected.

ISO/IEC 27001 – The international gold standard for information security

ISO/IEC 27001 sets the global benchmark for creating and maintaining an **Information Security Management System (ISMS)**, emphasizing the need for security measures that are both sufficient and balanced. This internationally acclaimed standard helps organizations proactively manage and protect their sensitive data through a systematic approach to risk management.

Controls in ISO/IEC 27001

Microsoft's Security Adoption Framework reinforces the latest revision of ISO/IEC 27001 by emphasizing *security hygiene*, which is a practice of continuous patching, regular posture assessments, and simplified tooling. This mindset shifts compliance from a one-time checklist to a continuous, proactive discipline.

Here's what's new in the latest version:

- **Updated Annex A controls**: The number of controls has been reduced from 114 to 93, reorganized into four themes to reflect modern security needs:
    - Organizational controls (37 controls)
    - People controls (8 controls)
    - Physical controls (14 controls)
    - Technological controls (34 controls)
- **New controls introduced**: The update includes **11 new controls** focusing on areas such as threat intelligence, data leakage prevention, and secure coding
- **Aligned with ISO/IEC 27002:2022**: The controls now mirror the guidelines provided in ISO/IEC 27002:2022 for better clarity and implementation



Annex A of ISO/IEC 27001 provides a detailed list of control objectives and controls, organized into 14 domains that cover various aspects of information security. Each domain includes multiple controls, totaling 114 in all.

ISO/IEC 27001:2022 Annex A – overview of controls

The 2022 update to ISO/IEC 27001 introduced a refined structure for Annex A, consolidating the controls into four main categories:

- **Organizational controls (A.5)**: Encompassing 37 controls, this category addresses policies, roles and responsibilities, risk management, and supplier relationships. Notable additions include threat intelligence, information security for use of cloud services, and ICT readiness for business continuity.
- **People controls (A.6)**: Consisting of 8 controls, this section focuses on personnel-related aspects such as screening, information security awareness, education and training, and remote working.
- **Physical controls (A.7)**: With 14 controls, this area pertains to the protection of physical assets and environments, including physical security perimeters, equipment maintenance, and secure disposal or re-use of equipment.
- **Technological Controls (A.8)**: Comprising 34 controls, this category covers technical measures such as secure authentication, data masking, and secure system architecture and engineering principles.

With this foundation in place, the next logical step is to translate these principles into a focused, actionable set of priorities. This is where the NCSE Top 10 items come in by providing a practical, high-impact checklist that helps organizations operationalize ISO 27001's controls in day-to-day security operations and make measurable progress fast.

The 10 Steps to cybersecurity

The NCSC has developed the 10 Steps to cybersecurity as a practical framework to help medium and large organizations understand and manage cyber risks at scale.

Introduced in 2012 and updated regularly, the 10 Steps to cybersecurity are designed not just for technical teams, but also for boards, executives, and business leaders who must embed cyber resilience into strategic decision-making. The framework encourages organizations to start with strong governance and risk management, then build layered defenses across technology, people, and processes.

Figure 2.5 highlights the foundations of the 10 Steps to cybersecurity:

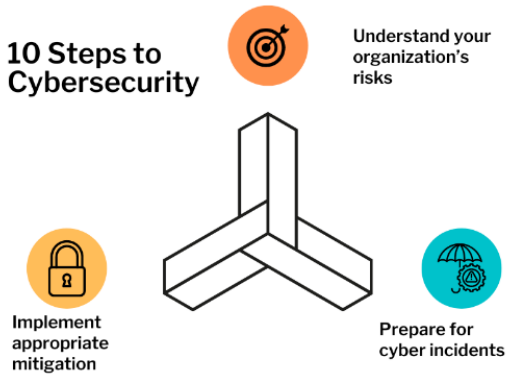


Figure 2.5: 10 Steps to Cyber Security by NCSC

Table 2.1 encompasses various domains of cybersecurity:

Step	Description
1. Risk Management	Establish clear leadership and accountability for cyber risk. Define policies, assess threats regularly, and ensure board-level oversight of cyber priorities.
2. Home and Mobile Working	Secure devices and connections used by remote workers. Apply VPNs, endpoint protection, and strong authentication policies.
3. User Education and Awareness	Educate staff to spot phishing, use strong passwords, and report suspicious activity. Cyber-aware teams are your human firewall.
4. Incident Management	Develop and regularly test an incident response plan. Include escalation paths, communication protocols, and recovery objectives.
5. Malware Prevention	Deploy multi-layered anti-malware defenses and keep them current. Segment networks to limit exposure if malware spreads.
6. Managing User Privileges	Enforce the principle of least privilege. Regularly review admin accounts and apply just-in-time access where possible.
7. Monitoring	Continuously log and analyze activity across your systems. Set up alerts for anomalies, unauthorized access, and suspicious behavior.
8. Network Security	Protect your network using segmentation, firewalls, and intrusion detection systems. Encrypt traffic and validate configurations.

Step	Description
9. Removable Media Controls	Restrict USB usage. Enforce encryption and scanning policies for removable devices to prevent data leakage and malware entry.
10. Secure Configuration	Harden systems by disabling unused features, applying patches promptly, and following vendor security baselines.

Table 2.1: Cybersecurity domains



Take a moment to explore the materials found in the URLs provided here as they will equip you with the tools necessary to chart your security journey:

- **NCSC GitHub repository**: Access a variety of tools, scripts, and guides on the NCSC GitHub repository to support your cybersecurity efforts: `https://github.com/ukncsc`
- **10 Steps to Cybersecurity Infographic by NCSC**: The NCSC's infographic provides a visual overview of each step, making it easy to understand and share across your organization: `https://www.ncsc.gov.uk/files/2021-10-steps-to-cyber-security-infographic.pdf`

General Data Protection Regulation (GDPR)

The **General Data Protection Regulation** (**GDPR**), introduced on May 25, 2018, transformed how organizations handle personal data across the **European Union** (**EU**). At its core, the GDPR is about giving individuals greater control over their personal information while creating a standardized framework that simplifies compliance for organizations operating in or interacting with EU citizens. The impact of the GDPR has reached far beyond Europe, influencing global data privacy practices and compelling any business dealing with EU residents' data to follow suit.

The GDPR's implications for cybersecurity

Although the GDPR is primarily a data protection regulation, its impact on cybersecurity is significant. It mandates the implementation of strong technical and organizational measures such as encryption of data at rest and in transit, strong access controls and MFA, regular patching and vulnerability management, network segmentation, logging and monitoring, secure backup and recovery to prevent unauthorized access, data leaks, and breaches. In the event of a data breach, organizations must report the incident to supervisory authorities within 72 hours, highlighting the importance of rapid detection and incident response.

One of the GDPR's cornerstone principles is *Data Protection by Design and by Default*. This means that privacy and security considerations must be built into systems and processes from the outset and not bolted on later. Organizations are expected to minimize data collection, enforce strict access controls, and ensure secure processing practices as standard.

Figure 2.6 outlines the essential steps to begin your GDPR compliance journey:

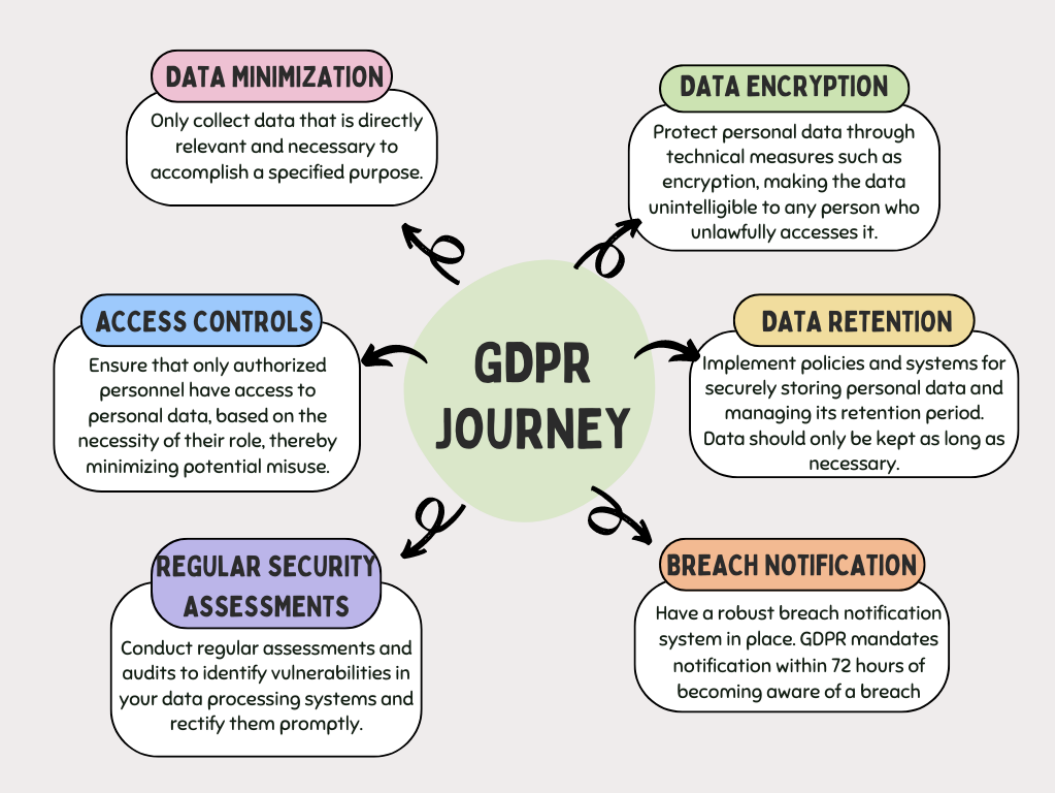


Figure 2.6: GDPR journey

In alignment with these requirements, organizations must document their data processing activities thoroughly, evaluate the impact of high-risk data processing, and implement strong vendor management practices to ensure that third-party partners meet GDPR standards. This rigorous approach to data security necessitates close collaboration with vendors and partners to verify that they adhere to similar levels of data protection. GDPR also strengthens individual data rights, mandating that organizations have mechanisms in place to manage requests for data access, correction, or deletion quickly and securely, creating an environment where personal data rights are actively respected and supported.



Useful resources to support your GDPR compliance can be found at the following links:

- **European Commission's GDPR Portal**: Comprehensive overview of GDPR rules, rights, and obligations under EU law: `https://ec.europa.eu/info/law/law-topic/data-protection/eu-data-protection-rules_en`
- **UK Information Commissioner's Office (ICO)**: Practical, detailed guidance on GDPR and broader data protection principles, tailored for organizations: `https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/`
- **National Cybersecurity Centre (NCSC) – GDPR Security Outcomes**: Actionable advice on aligning cybersecurity practices with GDPR compliance goals: `https://www.ncsc.gov.uk/guidance/gdpr-security-outcomes`

SOC 2

System and Organizations Controls Type 2 (**SOC 2**) is a globally recognized framework developed by the **American Institute of Certified Public Accountants** (**AICPA**). While it originated in the United States, its adoption is rapidly increasing among United Kingdom and European service providers.

SOC 2 assessments evaluate not only the design but also the operational effectiveness of an organization's controls over a defined period (typically 6–12 months). This provides a higher level of assurance than Type 1 assessments, which assess controls at a single point in time.

Here's why organizations should implement SOC 2:

- **Builds trust with clients**: Demonstrating SOC 2 compliance assures clients and stakeholders of the organization's commitment to safeguarding their data, enhancing trust and confidence in the services provided
- **Competitive advantage**: SOC 2 compliance can serve as a differentiator in the marketplace, especially when competing for clients sensitive to data security and privacy
- **Risk management**: Adhering to SOC 2's principles helps organizations identify and mitigate risks related to data security, availability, and confidentiality, leading to more robust data management practices

- **Regulatory compliance**: For companies operating or serving clients in jurisdictions where data protection laws are stringent, SOC 2 compliance helps ensure adherence to legal and regulatory requirements, avoiding potential fines and legal challenges
- **Operational excellence**: The process of achieving SOC 2 compliance encourages organizations to review and optimize their control environments, leading to improved operational efficiency and data handling practices

Fundamentally, SOC 2 compliance is not just about fulfilling a set of requirements. It's about embedding a culture of security and trust that permeates every aspect of an organization's operations, thereby securing its data, reputation, and future growth.

The Center of Internet Security (CIS)

The **Center for Internet Security** (**CIS**) is a nonprofit entity dedicated to protecting both private and public organizations from cyber threats. One of its key contributions to cybersecurity is the development of the CIS Controls and the CIS Benchmarks. The CIS Controls are a set of best practices and security measures designed to defend against the most pervasive cyberattacks. These controls are prioritized and actionable, offering organizations of all sizes a clear pathway to improve their cyber defense. The CIS Benchmarks are consensus-based, industry-recognized secure configuration guidelines for hardening computer systems, software, and networks.



Access the CIS Benchmarks for more than 25 vendors, offering configuration recommendations and security best practices, directly from `https://www.cisecurity.org/cis-benchmarks`.

Don't forget to check the CIS GitHub repository to access the latest security configurations and tools: `https://github.com/CISecurity`.

Core components of CIS – Controls and Benchmarks

The CIS Controls are a set of 18 critical security measures that organizations can implement to protect against common cyber threats. They are organized into three tiers: basic, foundational, and organizational. These controls span areas such as secure configuration, account management, and audit logging. They provide a clear path to strengthen defenses by ensuring that you have considered and addressed all essential security foundations. These controls are regularly updated to reflect the latest security insights and can be tailored to an organization's specific needs.

The CIS Benchmarks are detailed, consensus-based guidelines for configuring technology securely. Covering over 100 different technologies, from operating systems to cloud services and network

devices, these benchmarks serve as a blueprint for reducing vulnerabilities and minimizing the risk of attacks. With more than 25 vendor-specific benchmarks available, including for Microsoft, Amazon, and Google, CIS provides a valuable resource for hardening systems across diverse environments.

Two key examples are as follows:

- **CIS Microsoft Azure Foundations Benchmark**, which defines a secure baseline for Azure subscriptions, covering identity, networking, logging, and storage
- **CIS Amazon Web Services (AWS) Foundations Benchmark**, which provides hardening guidance for core AWS services such as IAM, CloudTrail, and S3 to reduce misconfigurations in cloud environments

CIS and Microsoft

In collaboration with Microsoft, CIS has seamlessly integrated its benchmarks into a variety of Microsoft products, enhancing both security management and compliance capabilities across the Microsoft ecosystem. This integration includes support for Windows, Windows Server, Microsoft 365, and Azure, providing organizations with guided, actionable steps to achieve secure configurations within these environments. For users, this means accessing high-security standards directly from within Microsoft's platforms without extensive manual adjustments.

Automated security assessments

Through Microsoft Defender for Cloud and Microsoft Purview, the CIS Benchmarks are applied automatically, allowing organizations to monitor their security posture in real time. These tools evaluate configurations against CIS standards, generating actionable insights and recommendations to improve security compliance. This automated assessment process not only streamlines regulatory adherence but also gives companies confidence that they are upholding best practices in line with CIS standards.

Securing Azure with the CIS Benchmarks

If you're running workloads in Microsoft Azure, the CIS Microsoft Azure Foundations Benchmark (v2.0.0) is a must-have resource. Developed by CIS, this benchmark provides step-by-step guidance to help you secure your Azure environment by covering key areas such as identity and access management, network controls, monitoring, and logging.

Microsoft has made it easier than ever to adopt these best practices. Microsoft Defender for Cloud now includes the latest CIS benchmark as part of its built-in compliance policies. This means you

can continuously monitor your environment and receive real-time alerts when configurations drift from the recommended standards.

Even better, you can automate compliance through Azure Policy, which includes policy definitions mapped directly to CIS controls. This allows you to apply guardrails across your Azure resources and maintain consistent security settings at scale.



To get started with CIS in Azure and M365, check out the following links:

- **CIS Microsoft Azure Foundations Benchmark**: `https://www.cisecurity.org/benchmark/azure`
- **CIS Microsoft 365 Foundations Benchmark**: `https://www.ncsc.gov.uk/files/2021-10-steps-to-cyber-security-infographic.pdf`

What framework should you align your organization to?

Selecting the right cybersecurity framework is about aligning about aligning your security strategy with your business goals, risk appetite, and industry obligations. Start by identifying any mandatory regulatory requirements based on your sector and geography. For example, United States healthcare organizations must comply with **HIPAA**, while those processing payment data must meet **PCI DSS**. In the EU, the **GDPR** remains key for personal data protection, and new mandates, such as **NIS2**, are raising the bar on critical infrastructure security. Choosing a framework that complements these requirements can simplify audits, strengthen defensibility, and provide a reliable foundation for long-term compliance.

Beyond legal requirements, your framework should reflect how your business operates. If you need a flexible, risk-based model, **NIST Cybersecurity Framework (CSF) 2.0** is a leading choice, especially with its new Govern function, which brings board-level oversight into cybersecurity planning. For globally minded enterprises seeking a structured certification path, **ISO/IEC 27001** offers a robust **Information Security Management System (ISMS)**. If you're looking for more tactical guidance, the **CIS Controls** deliver clear, prioritized actions that are especially effective for SMBs or resource-constrained teams.

Increasingly, organizations are layering these frameworks to cover both strategic governance and operational excellence. Many also integrate Microsoft-specific guidance such as the **Cloud Adoption Framework (CAF)** and the **Well-Architected Framework (WAF)** to align cloud security posture with business priorities.

However, before jumping into these two frameworks, we need to review an extremely important initiative from Microsoft called the **Secure Future Initiative (SFI)**.

The Secure Future Initiative (SFI) – a new foundation for trust

In response to the escalating sophistication and frequency of cyber threats, Microsoft launched the **Secure Future Initiative (SFI)** in November 2023. This multi-year, organization-wide endeavour signifies a transformative shift in Microsoft's approach to cybersecurity, emphasizing proactive defence mechanisms, cultural change, and technological innovation.

The SFI was conceived following critical evaluations of Microsoft's security posture, notably after incidents such as the Storm-0558 and Midnight Blizzard cyberattacks. These events highlighted vulnerabilities in Microsoft's systems and underscored the need for a comprehensive overhaul of its security strategies. By dedicating over 34,000 full-time engineers to this initiative, Microsoft has embarked on its most extensive cybersecurity transformation to date.

Figure 2.7 represents what SFI stands for, as well as its core principles.



Figure 2.7: SFI core tenets

SFI is anchored in three core principles that guide its implementation across all facets of Microsoft's operations:

- **Secure by design**: Integrating security considerations from the inception of product development, ensuring that security is a foundational aspect rather than an afterthought
- **Secure by default**: Ensuring that security features are enabled by default in all products and services, reducing the reliance on user intervention for security configurations.
- **Secure operations**: Maintaining robust security practices throughout the operational life cycle, including continuous monitoring, threat detection, and rapid response protocols

To operationalize its principles, SFI focuses on six strategic pillars:

- **Protect identities and secrets**: Enhance identity management systems to prevent unauthorized access, including the implementation of phishing-resistant credentials and stringent access controls
- **Protect tenants and isolate systems**: Implement strict isolation protocols for different systems and tenants to prevent lateral movement of threats within the network.
- **Protect networks**: Strengthen network security through comprehensive asset inventory, micro-segmentation, and rigorous access controls to minimize potential attack surfaces
- **Protect engineering systems**: Secure the software development life cycle by enforcing code integrity, limiting access to development environments, and adopting Zero Trust principles
- **Monitor and detect cyberthreats**: Deploy advanced monitoring tools and AI-driven analytics to detect and respond to threats in real time across all systems
- **Accelerate response and remediation**: Establish rapid response mechanisms to address vulnerabilities and incidents promptly, reducing the window of exposure

Taken together, these six pillars move SFI from a set of security aspirations to a practical operating model. To embed these principles consistently across new and existing workloads, they need to be wired into how the organisation plans, designs, and runs its cloud platforms. This is where Microsoft's CAF becomes essential.

Microsoft's CAF and Azure's WAF

Microsoft's CAF and Azure's WAF provide essential guidance for organizations looking to transition to the cloud efficiently and securely. While both frameworks target Azure environments, they focus on different stages and aspects of cloud adoption and architecture, offering complementary paths to help organizations leverage the full potential of Azure. The WAF complements the CAF by helping teams assess their architecture across five pillars: cost optimization, operational excellence, performance efficiency, reliability, and security. Microsoft recommends using the Well-Architected Review Assessment to evaluate technical debt, governance maturity, and architecture alignment with business needs.

CAF

The **Cloud Adoption Framework for Azure** guides organizations through a structured, life-cycle approach to cloud adoption, addressing critical aspects such as strategy, planning, preparation, and governance. CAF is organized into phases: Strategy, Plan, Ready, Adopt, Govern, Manage, and Secure. Each phase is designed to align cloud goals with business objectives. For example, the Strategy phase helps teams identify motivations for moving to the cloud and set measurable goals, while the Govern phase focuses on maintaining compliance and security policies as organizations scale. Each phase includes tools, best practices, and templates to support cloud architects, IT professionals, and decision-makers in executing a successful cloud migration and operation plan.



New in 2024–2025: CAF now includes guidance on AI governance, sustainability, and Zero Trust architecture alignment as part of Microsoft's broader SFI.

Figure 2.8 shows several key components, each designed to address different aspects of the cloud adoption journey:

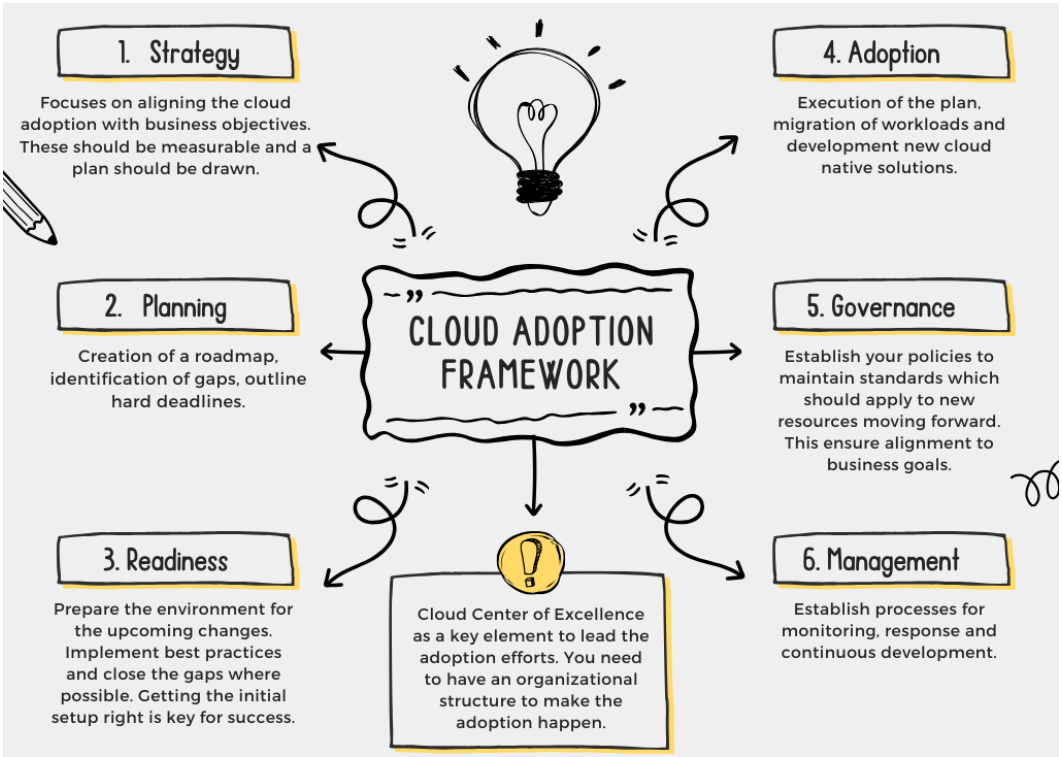


Figure 2.8: Components of Microsoft's CAF

To ensure a successful and sustainable cloud journey, organizations should approach cloud adoption through a structured framework. Let's a deeper look at the essential components:

1. **Strategy**: The foundation of any cloud journey, strategy ensures cloud adoption aligns with business goals. It defines why the organization is moving to the cloud by identifying key drivers, setting measurable outcomes, and outlining a high-level vision. A well-defined strategy ensures that cloud initiatives deliver real business value and remain aligned with broader organizational objectives.
2. **Planning**: In this phase, organizations translate strategy into action through a comprehensive adoption plan. This includes readiness assessments (across technology, people, and processes), skills gap identification, resource planning, and timeline development. Strong planning ensures clarity, accountability, and an executable path to the cloud.

3. **Readiness**: Before deployment begins, the organization prepares its cloud environment based on proven best practices. This includes configuring the environment for scalability, security, and compliance. Readiness ensures that workloads can be migrated or built with confidence and efficiency.
4. **Adoption**: This is the execution phase, where workloads are migrated to Azure or new cloud-native solutions are developed. Organizations use Azure's platform capabilities to meet business needs while optimizing performance, scalability, and cost-efficiency.
5. **Governance**: Governance helps maintain control and accountability as the cloud environment evolves. This includes defining policies, compliance standards, and budgetary controls to ensure consistent security, performance, and operational alignment throughout the life cycle.
6. **Management**: Effective management ensures post-adoption operational excellence. This involves establishing processes for monitoring, incident response, performance optimization, and cost control. Continuous improvement and transparency are key to long-term success in the cloud.
7. **Organization**: Organizational alignment is critical to cloud success. This component focuses on defining team roles, responsibilities, and operating models. Establishing a **Cloud Center of Excellence** (**CCoE**) helps promote best practices, drive cultural change, and provide centralized leadership for cloud adoption and operations.

The aim of this framework is to allow organizations to adopt Azure, taking a structured approach, ensuring that the transition delivers the anticipated business value and technical benefits while minimizing potential risks.



You can start your cloud journey by downloading the tools and templates for the CAF components from this link: `https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/resources/tools-templates`.

WAF

WAF provides a set of proven principles and best practices designed to help organizations build secure, high-performing, resilient, and efficient cloud architectures. Centered around five core pillars, the framework supports consistent decision-making throughout the life cycle of a workload from initial design to ongoing optimization.

These are the five pillars of the WAF:

- **Cost optimization**: Focused on maximizing value, this pillar emphasizes managing cloud spending by right-sizing resources, selecting appropriate pricing models, and monitoring usage. Regular reviews of resource consumption and allocation ensure that investments remain efficient and aligned with business goals, avoiding unnecessary costs.
- **Operational excellence**: This pillar supports the development of scalable and automated operations. It emphasizes the implementation of DevOps practices, monitoring, and incident response systems to maintain consistent and reliable operations. A strong feedback loop helps teams improve processes, reduce manual effort, and ensure agility in a dynamic cloud environment.
- **Performance efficiency**: Ensuring applications perform reliably under load, this pillar encourages the selection of the right compute, storage, and database resources. It also highlights the importance of designing for elasticity, with techniques such as horizontal scaling and load testing helping workloads adapt seamlessly to changing demands.
- **Reliability**: Reliability focuses on building resilient systems that can recover quickly from failures. It includes strategies for high availability, redundancy, disaster recovery, and fault tolerance. Leveraging Azure features such as auto-scaling and self-healing services helps minimize downtime and ensure business continuity.
- **Security**: Security is foundational to every workload. This pillar promotes a defense-in-depth strategy by combining identity and access management, encryption, threat detection, and compliance monitoring. Tools such as Microsoft Defender for Cloud and Azure Policy help organizations enforce security best practices and stay ahead of evolving threats.

To bring these pillars into practice, Microsoft provides the Azure Well-Architected Review, which is a self-assessment tool that helps organizations evaluate their cloud workloads against WAF best practices. It provides tailored recommendations that support continuous improvement and alignment with evolving business and technical needs.



Resources for the WAF can be found here:

- `https://learn.microsoft.com/en-us/azure/well-architected/pillars`
- `https://learn.microsoft.com/en-us/assessments/azure-architecture-review/`

CAF versus WAF – which one should you use?

Table 2.2 illustrates where and how CAF and WAF fit, depending on your organizational requirements:

Aspect	**CAF**	**WAF**
Focus	Strategy, governance, operations	Architecture design, workload optimization
When to Use	Planning or scaling your cloud journey	Designing or improving specific workloads
Audience	Cloud architects, business leaders	Developers, solution architects

Table 2.2: CAF versus WAF

Defining and implementing Zero Trust strategies for your infrastructure

In *Chapter 1* and throughout this chapter, we have discussed several cybersecurity solutions, frameworks, and best practices. By now, you have probably noticed one common denominator in most frameworks, which is **Zero Trust**. Zero Trust has become a defining cybersecurity model, challenging traditional perimeter-based security. By eliminating implicit trust and enforcing continuous verification, Zero Trust has evolved as a response to today's sophisticated cyber threats. From remote work environments to cloud computing and IoT, Zero Trust is built on a "*never trust, always verify*" principle that redefines secure access by assuming that no user or device, whether inside or outside the network, is automatically trusted.

The emergence of Zero Trust – a modern cybersecurity model

Zero Trust emerged to meet today's cybersecurity demands. Built on the principle of "*never trust, always verify*," it assumes that threats can come from anywhere, inside or outside the network. Instead of blanket trust, Zero Trust applies continuous validation of users, devices, and applications before granting access. This model leverages advanced technologies, including MFA, AI/ML-based behavioral monitoring, **Just-In-Time (JIT)** access, **Cloud Security Posture Management (CSPM)**, and **Endpoint Detection and Response (EDR)**.

Together, these tools provide real-time monitoring, context-aware access control, and rapid response to suspicious activity, which are crucial for defending hybrid and cloud-based infrastructures.



Did you know? The first cyber-attack

Surprisingly, one of the earliest examples of a cyber attack predates computers entirely.

In 1834, the French telegraph system used semaphore towers to relay messages. Brothers François and Joseph Blanc bribed a telegraph operator to insert errors into financial messages between Paris and Bordeaux. By interpreting the altered messages, they gained insider information and profited from the bond market, effectively exploiting the system for personal gain.

Though primitive by today's standards, the Blanc brothers' scheme highlights a key lesson: **where there is communication, there is vulnerability**. It's a reminder that security must always evolve with technology and threats.

From reactive to proactive — the shift in cyber defense

As cyber threats increase in frequency and complexity, organizations are moving from reactive defenses to proactive strategies. Regulations such as the GDPR and HIPAA have reinforced the need for preventative security measures and risk-based thinking. Modern cybersecurity goes beyond technology by incorporating governance, risk management, and compliance frameworks. Cybersecurity is now considered a board-level concern, with organizations investing in **Security Operations Centers** (**SOCs**), threat intelligence platforms, and AI-driven threat detection and response.

To combat emerging threats, organizations are deploying advanced tools such as encryption, which protects data in transit and at rest; cloud-native security, which addresses the unique risks of distributed, multi-tenant environments; and AI and machine learning, which enable rapid threat detection by analyzing massive datasets for suspicious behavior. These technologies enhance visibility, response time, and threat mitigation, especially in dynamic environments where traditional defenses fall short.

The disintegration of the network perimeter due to remote work, mobile devices, and cloud services has made Zero Trust not just a recommendation but a necessity. Unlike traditional models that trusted internal actors, Zero Trust enforces **continuous verification of every access request**, regardless of origin. It shifts the focus from *trust but verify* to *never trust, always verify.*



What does Zero Trust mean today?

The journey from physical locks to Zero Trust represents a seismic shift in how organizations approach security. No longer confined to firewalls and passwords, cybersecurity today is about **intelligent, adaptive defense** built on the assumption that threats are always present. By adopting Zero Trust, organizations gain a flexible, resilient framework built for the realities of the current and future threat landscape.

Core principles of Zero Trust

Let's now dive into the principles of Zero Trust.

Never trust, always verify

This principle challenges the traditional security model of *trust but verify* by adopting a more skeptical stance: no entity should be automatically trusted, regardless of its location inside or outside the organization's traditional security perimeter. Instead, every access request is subjected to rigorous authentication and authorization processes, with all communications encrypted to ensure security. This approach ensures that only validated and necessary interactions occur within the network, greatly reducing the attack surface.

Least privilege access

Zero Trust emphasizes least privilege access to limit the potential scope of a breach. Users and devices receive only the minimum permissions necessary for specific tasks so that even if an attacker gains access, their movement is restricted. This approach reduces lateral movement within the network, thereby protecting sensitive data and assets from unauthorized access. By confining permissions to the absolute essentials, Zero Trust architecture effectively minimizes the risk posed by compromised credentials or insider threats.

Assume breach

Operating under an *assume breach* mindset, Zero Trust treats potential intrusions as inevitable. This principle encourages continuous monitoring, rapid threat detection, and swift response protocols, all of which aim to identify breaches early, limit their impact, and isolate compromised areas to protect the broader network. By prioritizing detection and response, Zero Trust ensures that even if a breach occurs, its damage remains contained, and network integrity is maintained.

Micro-segmentation

Micro-segmentation divides a network into small, secure zones, each requiring individual authentication. This segmentation limits an attacker's ability to move freely within the network, as each segment enforces strict access controls. By isolating sensitive resources within these zones, organizations can contain potential breaches and reduce the overall impact of any single compromise. Micro-segmentation is essential in Zero Trust environments, where minimizing exposure and maximizing control are key goals.

Explicit verification

Zero Trust mandates explicit verification for every access attempt, utilizing MFA and advanced authentication techniques. By requiring continuous validation of both users and devices, Zero Trust decreases the likelihood of unauthorized access, securing sensitive data and critical infrastructure. MFA and explicit verification provide an added layer of security by ensuring that access is granted only to verified entities, thus fortifying the network against intrusion attempts.

With these concepts in mind, let's explore how to implement them.

Implementing a Zero Trust architecture

Creating a **Zero Trust architecture** involves more than just technical upgrades. It requires a comprehensive approach combining technology, policy, and education. We will now discuss the essential components of implementing Zero Trust.

Identity verification

The backbone of Zero Trust is **Identity and Access Management (IAM)**, which enforces strict identity verification for every access request. This includes implementing **Multi-Factor Authentication (MFA)**, **Single Sign-On (SSO)**, and **contextual authentication**. By verifying users with multiple identity checks and evaluating contextual factors such as location, device, and role, IAM solutions ensure that only authorized individuals gain access. This adaptive, risk-based approach reduces the chance of unauthorized access by validating identities with multiple signals before granting access to any resources.

Device security

Zero Trust architecture requires ensuring that all devices accessing the network meet stringent security standards. This means assessing devices for malware, ensuring compliance with organizational security policies, and checking for recent security patches. **Endpoint security solutions** play a critical role here by protecting against potential vulnerabilities and malware that could compromise network integrity. By regularly assessing device health, organizations can keep the network secure and prevent compromised devices from introducing risks.

Network security

Network security within Zero Trust hinges on **micro-segmentation**, which involves dividing the network into isolated zones, each with separate access controls. This segmentation minimizes the ability of attackers to move laterally if they gain access to one part of the network. **Data encryption** in transit and at rest, combined with rigorous network access controls and encrypted traffic inspection, ensures that all communications remain secure and inaccessible to unauthorized users. This layered approach to network security strengthens protection against external threats and insider risks alike.

Data security

Classifying and securing data based on its sensitivity is a core element of Zero Trust. By applying **stringent access controls** and encrypting data both at rest and in transit, organizations can safeguard sensitive information. Access is only granted to individuals with explicit authorization, significantly reducing the likelihood of data breaches. Data security policies and automated monitoring further enhance protection, ensuring that sensitive information is only accessible by authorized personnel under secure conditions.

Security policies and governance

A Zero Trust model is only as effective as the policies that support it. Comprehensive security policies that reflect Zero Trust principles should be developed and enforced across the organization. This includes continuous monitoring, logging, and analyzing network and user activities to detect and respond to threats swiftly. **Governance frameworks** ensure these policies are embedded in the organizational culture, creating accountability and a security-first mindset at all levels.

Awareness and training

An informed workforce is a critical asset in a Zero Trust framework. Regular training on identifying phishing attacks, secure internet practices, and the role of employees in cybersecurity strengthens the overall security posture. Employees who understand the principles of Zero Trust and their role in upholding it act as an additional line of defense, reducing the likelihood of human error that could lead to a security breach.

Challenges and considerations in Zero Trust implementation

Adopting a Zero Trust model presents a transformative opportunity for enhancing organizational security. However, its implementation is not without challenges, necessitating thoughtful consideration and strategic planning. Let's look at some of the primary challenges that organizations face when moving to a Zero Trust framework.

Legacy systems

One of the most significant obstacles to the adoption of Zero Trust is the integration of legacy systems that may lack the capabilities needed for modern Zero Trust requirements. Many older systems cannot support sophisticated identity and access protocols, such as continuous authentication or advanced endpoint verification. Retrofitting these systems to align with Zero Trust can be complex and costly, often requiring custom solutions or even system overhauls. This challenge is particularly pronounced in industries with specialized, long-standing equipment or software, as updating these systems can also lead to substantial operational disruptions.

Cultural shift

Implementing Zero Trust is not just a technical shift. It's also a cultural one. Traditional perimeter-based models instill a sense of implicit trust within network boundaries, whereas Zero Trust requires **continuous verification** of all users and devices, regardless of location. This shift demands that all employees, from IT teams to end-users, adopt a mindset of vigilance. Organizations may need to invest heavily in **training and awareness programs** to help staff understand their roles in this new security model. Cultivating a security-first mindset across all departments is critical for Zero Trust's success and requires ongoing support to maintain compliance and reinforce best practices.

Cost and complexity

The financial and technical complexity of deploying Zero Trust can be daunting. Implementing this model often necessitates investments in new technologies, such as IAM systems, network segmentation solutions, and continuous monitoring tools. Additionally, restructuring network architectures to support Zero Trust is a complex endeavor, requiring skilled cybersecurity and network architecture expertise. For many organizations, especially small and mid-sized businesses, the costs associated with these upgrades can be prohibitive. Careful planning and phased implementation are crucial to prevent overwhelming budgets and resources.

Integration with existing security measures

Another challenge lies in aligning Zero Trust with an organization's existing security infrastructure. Many companies have accumulated a patchwork of security solutions over time, and integrating these disparate systems within a cohesive Zero Trust framework requires careful planning. For instance, existing firewalls, antivirus programs, and VPN solutions may need significant reconfiguration or replacement to comply with Zero Trust principles. Successful integration often involves updating security policies and ensuring interoperability between legacy solutions and modern Zero Trust technologies.

Regulatory compliance

As organizations adopt Zero Trust, they must also ensure that their practices align with regulatory requirements and data protection laws. Implementing continuous verification and data segmentation can be complex in environments regulated by stringent privacy laws such as the GDPR or HIPAA. Organizations need to verify that Zero Trust protocols do not inadvertently conflict with these legal requirements and must often conduct compliance audits throughout the transition. Achieving both compliance and comprehensive security can be challenging, but it is essential to prevent legal repercussions and maintain customer trust.

Despite all the challenges, implementing Zero Trust is a strategic move that aligns with modern cybersecurity needs, offering substantial long-term benefits in security, compliance, and efficiency. While implementation requires a deliberate, strategic approach, the gains in resilience and data protection make Zero Trust a compelling option for organizations of all sizes.

Planning your Zero Trust strategy

Implementing a Zero Trust architecture requires a detailed understanding of the organization's unique environment, a clear identification of what needs protection, an analysis of data flow, and the development of a comprehensive policy and roadmap.

Let's break down the steps of planning a Zero Trust strategy:

Understanding your environment

Understanding your environment is essential before implementing Zero Trust because you can't protect what you don't know exists or how it behaves. You need a clear view of your identities, devices, applications, data flows, and dependencies so that access policies, segmentation, and controls are designed around reality and not guesswork. Without this foundation, Zero Trust risks breaking critical services while still leaving high-value assets exposed.

You first need to identify and classify your assets:

- **Inventory of assets**: The first step involves creating a detailed inventory of all assets within the organization, covering both hardware (servers, endpoints, network devices) and software (applications, operating systems, data storage solutions).
- **Classification**: After creating an inventory of your assets, classify them based on criticality and sensitivity. This prioritization helps to determine the necessary level of protection for each asset, focusing efforts where they are most needed.

Next is a data flow analysis:

- **Mapping data flows**: A thorough understanding of how data moves within and outside the organization is crucial. Without understanding how data enters, moves, and leaves your organisation, you can't see where it's at risk. Clear data-flow visibility lets you place the strongest controls around sensitive information, detect risky or unexpected movements, enforce precise Zero Trust and DLP policies, and demonstrate compliance with confidence. This step includes identifying data entry, storage, and exit points.
- **Identifying sensitive data**: Pay particular attention to locating sensitive data, understanding its processing pathways, and identifying access controls. This information is vital for applying the correct security measures. In a payroll system, you would first locate all instances of sensitive data, such as employee salaries, bank details, and National Insurance numbers (HR database, exported CSV files, reporting tools, backups). You'd map how that data is processed—who enters it (HR), which systems consume it (payroll engine, finance), and where it's transmitted (for instance, to the bank or to HMRC, if it's in the UK). Finally, you'd review who has access at each step (HR staff, finance, IT admins, external processors) and what controls are in place (role-based access, MFA, encryption, DLP). This end-to-end view lets you apply the right security measures, such as tighter **role-based access control (RBAC)**, encryption in transit/at rest, and stricter sharing controls exactly where the sensitive data is most exposed.

Then, we identify the protect surface:

- **Defining the protectsurface**: Focus on the most critical **data, applications, assets, and services (DAAS)** that require the highest security level. In a healthcare organization, you might define the protect surface as electronic patient records, the clinical apps used to access them, the underlying database servers, and the VPN/identity services that control access. Rather than trying to secure everything at once, you prioritize these DAAS elements for the strictest controls, such as strong MFA, network micro-segmentation, encryption, and continuous monitoring, because compromise here would have the highest impact on patient safety, privacy, and regulatory compliance.
- **Focus on critical assets**: Concentrating on the critical assets identified earlier enables more efficient resource and security measure allocation.

Next is mapping the transaction flows:

- **Understanding user and device behavior**: Analyze normal access patterns and behaviors of users and devices interacting with the network and applications
- **Identifying legitimate access patterns**: Establishing baseline behaviors is crucial for detecting anomalies that may indicate security threats

Then we create a Zero Trust policy and roadmap:

- **Defining security policies**: Develop comprehensive security policies based on Zero Trust principles, including identity verification, access control, and threat detection and response
- **Incorporating stakeholder input**: Ensure the policies align with business needs and compliance requirements by engaging with stakeholders across the organization

These are the implementation phases:

- **Phased approach**: Adopt a phased approach to implementation, prioritizing areas of highest risk or criticality
- **Pilot programs**: Start with pilot programs to test Zero Trust controls and policies on a small scale, facilitating adjustments before broader application

Monitoring and adjusting are important:

- **Continuous monitoring**: Utilize tools and processes for ongoing monitoring of network traffic, user behavior, and device health to adhere to Zero Trust policies

- **Iterative improvement**: Leverage insights from monitoring and incident responses to continuously refine and enhance Zero Trust policies and controls

In this section, we've explored the core principles, policies, and implementation strategies of the Zero Trust model. Now, we'll delve into the technologies and security solutions available in the market that support and enhance Zero Trust architectures.

Exploring Zero Trust technologies and solutions

Protecting modern infrastructure requires a layered approach with technologies that continuously verify the identities, devices, and behaviors of users within a Zero Trust framework. We'll now discuss the key technologies that support a Zero Trust model, focusing on their roles in securing today's dynamic, hybrid environments.

Identity and Access Management (IAM)

IAM is central to Zero Trust. It enables robust identity verification through policies that ensure only authenticated, authorized users can access necessary resources. IAM solutions manage user identities such as accounts, devices, and systems while controlling permissions across the organization. Core IAM components include MFA, SSO, and RBAC, all of which reduce unauthorized access risks and streamline secure access for legitimate users.

MFA adds an extra layer of security by requiring multiple verification factors, significantly lowering the risk posed by compromised credentials. SSO improves security and user experience by allowing users to access multiple applications with a single set of credentials. Contextual and adaptive authentication methods adjust requirements based on contextual factors such as location and device health, providing flexible security while enhancing usability.

Endpoint security

Endpoint security protects all endpoint devices, including laptops, smartphones, and desktops that are connected to the corporate network. Endpoint security prevents these devices from becoming potential entry points for attackers. Endpoint solutions include antivirus software, intrusion detection systems, and more advanced tools such as **Endpoint Detection and Response (EDR)** and Zero Trust application controls. Key areas of endpoint security include the following:

- **EDR**: EDR continuously monitors endpoints for signs of malicious activity, enhancing detection, investigation, and response capabilities

- **Device health checks**: Before access is granted, endpoints undergo checks to verify compliance with security standards, including up-to-date patches, antivirus software, and secure configurations

Network security

Network security is essential for Zero Trust as it uses strategies such as micro-segmentation to divide networks into isolated zones, each with its own access requirements. Encryption of data in transit and at rest, along with robust network access controls, ensures that only authenticated communications can pass.

A critical component of network security in Zero Trust is micro-segmentation, which divides the network into secure segments to help contain threats and prevent lateral movement. Encrypting data ensures that even if intercepted, the data remains unreadable without the proper decryption keys. Zero Trust Network Access (**ZTNA**) controls access to applications and services based on user identity and risk level, often replacing traditional VPNs for remote access.

Data security and encryption

In Zero Trust, data security and encryption protect sensitive information from unauthorized access and breaches. Encryption keeps data secure in transit and at rest, while data classification and **Data Loss Prevention** (**DLP**) policies help monitor and control the flow of sensitive data. DLP classifies data by sensitivity, using DLP policies to prevent unauthorized access or transfer of critical information.

Security policies and governance

Security policies and governance are the backbone of Zero Trust, setting guidelines that direct how security measures are implemented and maintained. Governance aligns security practices with organizational goals and ensures compliance with regulatory standards. Important elements include policy orchestration and automation, security information, and event management. Policy orchestration and automation automate policy enforcement to reduce the risk of human error and ensure consistent security practices security information and event management (SIEM) solutions aggregate and analyze log data from across the organization to detect potential security incidents and facilitate rapid response.

Zero Trust implementation is a multi-dimensional process, requiring a strategic blend of IAM, endpoint security, network controls, data security, and governance. In the next section, we'll explore leading vendors and solutions, with a focus on Microsoft's tools for Zero Trust integration.

How can you achieve Zero Trust with Microsoft solutions?

Microsoft provides a robust suite of solutions tailored to support Zero Trust architectures, focusing on identity management, device security, data protection, network controls, and continuous threat monitoring. Here's how Microsoft's integrated tools help achieve a Zero Trust model across these pillars:

- **IAM with EntraID**: Microsoft Entra ID (formerly Azure AD) forms the foundation for identity management in a Zero Trust framework. Key features include SSO for seamless access across applications and MFA to ensure strong identity verification. Entra also supports conditional access policies, which assess contextual factors such as location, device compliance, and risk level before granting access, allowing more dynamic security that aligns with Zero Trust principles.
- **Device security with Microsoft Endpoint Manager**: To secure endpoints, Microsoft Endpoint Manager combines **Intune** and **Configuration Manager** for centralized management and compliance checks on Windows, iOS, Android, and macOS devices. Through EDR powered by Microsoft Defender for Endpoint, organizations gain advanced detection and response capabilities, which are crucial for identifying and mitigating threats on devices before they impact the network.
- **Data protection with Microsoft Information Protection**: Data protection is essential in Zero Trust, and Microsoft Purview offers tools such as **Data Classification** and **DLP** policies across Microsoft 365 services such as Teams, SharePoint, and Exchange. Purview automates data classification, enabling organizations to discover and govern sensitive information. These tools help ensure that only authorized users can access or share sensitive data, maintaining robust data protection across various environments.
- **Network security with Azure Network Security**: Azure's network security tools, such as **Azure Firewall** and **Azure Front Door**, enhance network segmentation and secure data flow. Azure Firewall provides high-availability, stateful firewall capabilities, while Azure Front Door offers web application firewall services to protect internet-facing applications. Together, these tools enforce rigorous network access control, a key aspect of reducing lateral movement within a Zero Trust framework.
- **Zero Trust network access with Entra ID application proxy and Azure Virtual WAN**: Microsoft's Entra ID application proxy enables secure, identity-driven access to on-premises applications without requiring a broad network opening, supporting the Zero Trust principle of minimal access. **Azure Virtual WAN** complements this by creating a global network transit architecture, integrating with Azure Firewall and VPN services to protect hybrid and multi-cloud environments.

- **Visibility, compliance, and automated response**: **Microsoft Sentinel** and **Microsoft Defender for Cloud** bring together visibility, threat intelligence, and automated responses to detect and mitigate threats in real time. Sentinel, Microsoft's cloud-native SIEM, aggregates threat signals across the ecosystem and automates incident response, allowing quicker and more effective responses. Defender for Cloud extends these capabilities to hybrid and multi-cloud environments, ensuring security posture management and threat protection are integrated across all resources.



Lessons from the field

Remember, Microsoft's ecosystem is designed to be integrated, which means that these solutions work together to provide a seamless and comprehensive Zero Trust security model. Engaging with Microsoft's documentation, support, and community can also provide additional insights and best practices for implementing Zero Trust with their solutions. Most importantly, documentation and features constantly change. Always refer to Microsoft Learn docs to stay up to date.

The future of Zero Trust

The evolution of Zero Trust is inseparable from the rapid advancement of technology, the shifting cyber threat landscape, and the ongoing transformation of organizational structures. As we look ahead, Zero Trust must continue to adapt, placing innovation at the heart of its progression.

Emerging technologies such as **quantum computing** and **AI** present both challenges and opportunities. Quantum computing threatens to disrupt current encryption standards, making the development of **quantum-resistant cryptographic algorithms** a critical priority. Simultaneously, innovations such as **quantum key distribution (QKD)** offer promising new ways to achieve unbreakable communication channels, aligning perfectly with Zero Trust's foundational principle of continuous verification.

Meanwhile, AI and machine learning are becoming indispensable to Zero Trust strategies. Their ability to deliver **predictive insights** and automate **real-time security policy adjustments** enables organizations to respond to threats dynamically. As AI continues to evolve, it will enhance the detection of anomalies, reduce response times, and streamline policy enforcement, further embedding Zero Trust as a proactive, intelligence-driven security model.

Looking forward, Zero Trust is not only a cornerstone of current cybersecurity strategies; it will also shape the ethical and practical implementation of future technologies.



Lessons from the field

The **Microsoft Sentinel Zero Trust (TIC 3.0)** solution, available in the Microsoft Sentinel Content Hub, provides security teams with a robust framework for aligning their operations with both Zero Trust principles and the **Trusted Internet Connections (TIC) 3.0** model—a United States federal cybersecurity initiative focused on securing modern enterprise environments. This solution delivers comprehensive monitoring, analytics, and automated response capabilities that span over 25 Microsoft and third-party products. It includes the following:

- A **Zero Trust Workbook** for visualizing posture across core pillars
- **Analytics Rules** to detect suspicious activity in alignment with Zero Trust strategies
- **Automated Playbooks** to respond to incidents faster and more consistently

It supports a variety of deployment environments, including **cloud, multi-cloud, hybrid**, and **on-premises** infrastructures, helping organizations meet compliance and security requirements more effectively.

To get started, only Microsoft Sentinel and Microsoft Defender for Cloud are required, with optional integrations that enhance functionality through services such as Microsoft Defender for Endpoint, Microsoft Entra ID, and Microsoft Purview.

Learn more here: `https://learn.microsoft.com/en-us/azure/sentinel/sentinel-solution`.

Now that we've covered the foundations and significance of Zero Trust, it's time to take action. How can you confirm that your Microsoft tenant aligns with essential Zero Trust principles? Fortunately, there are free assessment tools developed by security experts and agencies to help evaluate your tenant's security posture. They are mentioned later in the book!

Summary

In this chapter, we explored foundational cybersecurity frameworks and best practices, with a strong focus on the Microsoft **Cloud Adoption Framework (CAF)** and the Azure **Well-Architected Framework (WAF)**. These frameworks provide structured guidance for organizations adopting cloud technologies, helping to optimize architecture, security, and operational performance in the cloud.

We also examined leading cybersecurity standards, including NIST CSF, NCSC CAF, Cyber Essentials, ISO/IEC 27001, CIS Controls, GDPR, and SOC 2. Each framework offers a unique approach to improving security posture and aligning with regulatory and industry-specific requirements.

A major focus of this chapter was the Zero Trust security model, which is a foundational principle in modern cybersecurity and DevSecOps strategies. We covered its core tenets, such as *never trust, always verify*, least privilege access, micro-segmentation, assume breach, and continuous validation. Practical guidance was also provided for implementing Zero Trust, including strategies around identity and access management, device and network security, data protection, governance, and user education.

But before we move into technical implementation, it's important to recognize that cybersecurity success isn't just about tools; it's also about partnerships. In the next chapter, we'll shift to Microsoft's Security Partner ecosystem. You'll learn how to leverage the Microsoft Partner Center, CPOR, and PAL, explore security specializations, and discover how the Microsoft AI Cloud Partner Program is empowering the next wave of secure innovation.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



3

Microsoft Security for Partners

The Microsoft Partner Center is the central platform for businesses that collaborate with Microsoft, offering tools to manage memberships, access benefits, track competencies, and engage with customers more effectively. As Microsoft evolves toward an AI-driven future, Partner Center plays a critical role in enabling security-focused partners to build, deploy, and scale modern solutions.

In this chapter, we'll explore how Microsoft's AI Cloud Partner Program empowers organizations to integrate advanced security services into their offerings. We'll unpack the structure of this program, highlight the advantages of achieving a Solutions Partner designation, and outline the strategic benefits of earning advanced specializations. These distinctions not only elevate your visibility within Microsoft's ecosystem but also unlock co-selling opportunities, incentives, and technical support.

We'll also examine key programs designed specifically for security partners, including the Microsoft-certified MXDR initiative, and break down the mechanics behind **Claiming Partner of Record** (**CPOR**) and **Partner Admin Link** (**PAL**), the two critical tools for ensuring proper recognition for driving Microsoft security consumption.

By the end of this chapter, you'll understand how to navigate and maximize the Microsoft Partner landscape to drive stronger engagement, gain recognition for your influence, and grow your security business with Microsoft.

This chapter covers the following key topics:

- The Microsoft AI Cloud Partner Program
- Exploring security-focused partner programs
- Microsoft Cybersecurity Investments Program

- Partner security: protecting your customers
- Get recognized by Microsoft for your success: CPOR and PAL

The Microsoft AI Cloud Partner Program

The **Microsoft AI Cloud Partner Program** is designed to empower partners who are building and delivering AI-driven solutions using Microsoft's industry-leading platforms. This program equips partners with a full suite of enablement resources, including technical training through Microsoft Learn, AI-focused developer workshops, and hands-on guidance to build solutions with Security Copilot and generative AI. Whether you're enhancing existing services or launching new offerings, the program ensures that you have the support needed to succeed. It reflects Microsoft's commitment to accelerating innovation across its ecosystem, providing access to powerful tools such as Azure OpenAI, Azure AI, and comprehensive machine learning capabilities.

Beyond technical support, the program includes robust go-to-market benefits such as sales enablement, co-selling opportunities, and marketing resources, which help partners grow their reach and scale their AI business. It also encourages knowledge-sharing and collaboration through community events and Microsoft's global partner network.

The program is inclusive of a wide range of partners, including **independent software vendors (ISVs)**, **managed service providers (MSPs)**, system integrators, and advisory firms. Each participant can chart a personalized path to integrating AI across cloud, data, and security workloads.

As Microsoft continues to lead the charge in responsible and secure AI, the AI Cloud Partner Program stands as a strategic opportunity for partners to future-proof their offerings, deepen their Microsoft engagement, and differentiate themselves in an increasingly competitive market.

Benefits of joining the Microsoft AI Cloud Partner Program for security providers

For security-focused businesses, joining the Microsoft AI Cloud Partner Program unlocks a significant competitive edge. Partners will gain privileged access to Microsoft's most advanced security technologies, such as Microsoft Defender XDR, Sentinel, Entra, and Security Copilot, alongside the training, certifications, and technical support needed to deploy them with confidence and expertise.

Achieving the Security Solutions Partner designation within the program validates a partner's thorough capability in safeguarding customers with integrated identity, threat protection, and compliance solutions. It signals to the market that your organization meets Microsoft's high standards for performance, skilling, and customer success.

Beyond recognition, the program offers the following practical advantages:

- **Strategic technical guidance** and access to Microsoft engineers
- **Go-to-market support** to help scale offerings and accelerate growth
- **Private Preview access** to emerging security products and features
- **Eligibility for incentives** and co-sell engagements with Microsoft's sales teams

Security Partners also benefit from direct connection to a global network of peers, industry experts, and Microsoft product groups. This fosters collaboration, innovation, and faster time-to-market, ensuring that you stay ahead of evolving threats and customer expectations. By aligning with Microsoft's security vision and participating in the AI Cloud Partner Program, your organization positions itself at the forefront of modern cybersecurity, ready to meet complex customer needs with best-in-class solutions.

Becoming a member of the Microsoft AI program

The process to become a member of the Microsoft AI program is straightforward. To join the program, follow these steps:

1. **Create your Microsoft account**: Start by setting up a Microsoft account if you don't already have one. This will be your entry point to access Microsoft services, including Partner Center platform.
2. **Sign up at the Microsoft Partner Center**: Visit the Microsoft Partner Center website and sign up using your Microsoft account. Partner Center is the platform through which you'll manage your partnership, access resources, and more. You can access Partner Center by following this link: `https://partner.microsoft.com/`.
3. **Complete your partner profile**: Fill in your organization's details to complete your partner profile. This information is crucial for Microsoft to understand your business and for potential customers to find you through Microsoft's partner directories.
4. **Choose the right membership level**: Microsoft offers several types of membership, including a free Network Partner level and paid options tied to solution area designations. Assess your business objectives and technical maturity to select the membership path that aligns best with your goals.
5. **Enroll in the Microsoft Partner Network**: Follow the prompts to enroll in the Microsoft Partner Network. This may involve accepting legal agreements and, depending on your chosen membership level, paying any associated fees.

6. **Complete the business verification process**: Microsoft will verify your business to ensure that it meets the requirements for partnership. This process varies by country but generally includes verifying your business's legal existence and operational status.
7. **Unlock partner benefits**: Once your enrollment is approved and your business is verified, you'll gain access to a broad suite of benefits. These include training resources, solution playbooks, technical support, and sales and marketing enablement.

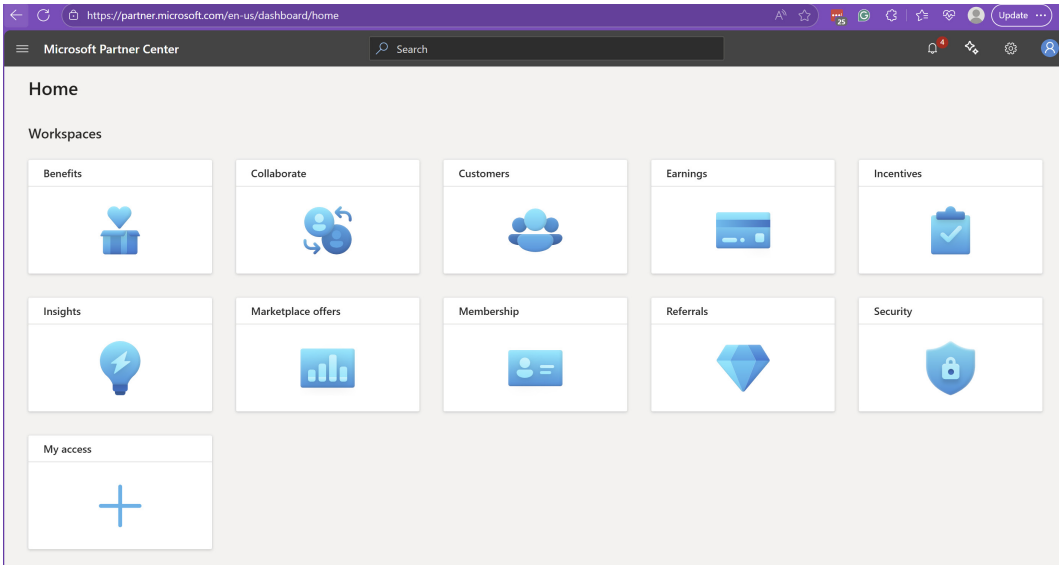


Figure 3.1: Microsoft Partner Center home – full access

Congratulations, you are now a member! The next step is to become a Solutions Partner.

Becoming a Microsoft Solutions Partner

The Microsoft Solutions Partner designation is a trusted marker of excellence that showcases an organization's technical expertise and proven success in delivering Microsoft Cloud solutions. It spans six key solution areas: Security, Modern Work, Infrastructure (Azure), Digital & App Innovation (Azure), Data & AI (Azure), and Business Applications. This designation sits at the core of the Microsoft AI Cloud Partner Program and is designed to help customers identify partners who can deliver impactful, industry-relevant outcomes through advanced cloud capabilities.

To attain a Solutions Partner designation, an organization must meet specific criteria such as certifications or Azure revenue in performance, skilling, and customer success. Performance is measured by the ability to expand the customer base, skilling is quantified by the certifications earned by the organization's members, and customer success is evaluated by the growth in usage

and deployment of Microsoft products among customers. The program provides flexibility for partners to chart their path to qualification but requires a demonstration of high-quality service to customers. This ensures that customers looking for partners can trust the Solutions Partner designations.

The Solutions Partner program offers different paths for qualification, recognizing that partners may have diverse growth strategies and work with customers of varying sizes. For instance, in the *Modern Work* and *Business Applications* areas, there are distinct paths for enterprises and **small and medium businesses (SMBs)** with different thresholds and criteria to earn points for the designation.

Partners are assessed on a scoring system and must meet the requirements within a qualification eligibility window to either enroll or renew their designation. The partner capability score is a composite score that measures a partner's performance in these categories and is used to determine eligibility for the Solutions Partner designation.



Lessons from the field

If you're looking to showcase or simulate Microsoft security solutions in real-world scenarios, the customer digital experience (CDX) provides hands-on labs, immersive demos, and preconfigured environments. It's a powerful tool for partners seeking to explore Microsoft Defender XDR, Microsoft Sentinel, and Dynamics in action: `https://cdx.transform.microsoft.com/`.

Let's explore the options available for security partners next.

Solutions Partner for Security

The **Solutions Partner for Security** designation is one of Microsoft's most impactful partner recognitions, highlighting an organization's capability to help customers protect their environments with integrated security, compliance, and identity solutions across Microsoft 365 and Azure. This designation isn't just a badge; it's a signal to customers and Microsoft sellers that your organization meets a high standard in delivering secure, scalable, and effective cloud solutions.

To attain this designation, your organization must meet thresholds in Microsoft's **partner capability score**, which is comprised of three dimensions:

- **Performance**: Measured by the number of new customers onboarded and the depth of Microsoft product engagement

- **Skilling**: Assessed by the number of individuals in your organization who have earned Microsoft security certifications, such as SC-100, SC-200, AZ-500, or MS-500
- **Customer Success**: Based on active usage and consumption of Microsoft security solutions by your customer base, including metrics such as **monthly active users** (**MAU**) and **Azure consumed revenue** (**ACR**)

Microsoft provides visibility into your current score and progress via Partner Center, and you must maintain your eligibility to retain the designation annually.



Use the Microsoft Partner dashboard to track your standing and access improvement recommendations: `https://partner.microsoft.com/`.

Solution designation for SMB

As Microsoft continues to refine its partner ecosystem, the Solutions Partner for Security designation now includes a dedicated SMB path, enabling partners to demonstrate security competency specifically within environments ranging from 5 to 300 seats. This designation provides a structured way to validate your ability to deliver Microsoft Security solutions to SMBs using Defender XDR, Microsoft Sentinel, and Microsoft 365.

This section summarizes what is required to attain the SMB designation, how it applies to multitenant security operations, and how partners can align their service delivery model to meet Microsoft's criteria across performance, skilling, and customer success.

Designation framework

The SMB path follows the same three-category structure as the enterprise designation but is tailored to a small and medium-sized customer base:

Category	SMB requirements	Max points
Performance	Up to five net new SMB tenants with M365 or Azure Security	20
Skilling	Two prerequisite certs + up to six scoring certs	40
Customer success	Usage growth and successful deployments	40

Table 3.1: SMB security designation requirements

A minimum of **70 points** across these categories is required to attain the designation.

Performance: growing your customer base

Partners earn points by onboarding net new SMB tenants using Microsoft 365 Security (5–300 seats) or Azure Security workloads generating at least $600 in ACR over the trailing 12 months. Each new customer counts toward a maximum of 20 points.

It is best practice to use Azure Lighthouse and CSP to streamline onboarding and attribution. Ensure that PAL or CPOR associations are in place to recognize customer ownership in Partner Center.

Skilling: building certified expertise

To qualify under the SMB path, partners must have team members certified in Microsoft security technologies:

- Prerequisites (earn points):
    - AZ-500: Azure Security Engineer Associate
    - SC-200: Security Operations Analyst
- Scoring certifications (6.67 pts each, up to 6 people):
    - SC-100: Cybersecurity Architect Expert
    - SC-300: Identity and Access Administrator
    - SC-400: Information Protection and Compliance Administrator



Certification Weeks for Partners offer streamlined, instructor-led training aligned to these exams.

Customer success: driving usage and deployments

This category measures both usage growth and active deployments among SMB customers:

- Usage growth:
    - Microsoft 365: 1 point per 50-seat increase (max 1,000 seats)
    - Azure Security: 1 point per $750 ACR growth (up to $15,000)
- Deployments:
    - Up to 6 net new deployments, defined as SMB tenants with active M365 Security usage or $750–$12,000 ACR in Azure Security

Use Microsoft Sentinel's Content Hub and Workspace Manager to deploy detection content, activate analytics, and ensure each tenant has baseline security configurations in place.

Aligning with your MSSP or multitenant model

The SMB designation complements multitenant delivery models using the following:

- **Microsoft 365 MTO** for cross-tenant identity, policy, and collaboration
- **Defender XDR** for centralized incident management
- **Microsoft Sentinel** for cross-tenant visibility and detection
- **Azure Lighthouse** for delegated access and automation at scale

Strategic value

Benefits of earning the SMB Security designation include Azure bulk credits and dev/test environments, Microsoft co-sell eligibility and marketplace placement, access to go-to-market services, sales consultations, and technical presales support, and recognition via customer-facing Microsoft badges.



For full benefits, visit `https://aka.ms/Solutionspartner.Benefits`.

Security advanced specializations

Once your organization earns the Solutions Partner for Security designation, you become eligible to pursue **Microsoft Security advanced specializations**. These are designed to validate advanced, scenario-based technical expertise and give your organization higher visibility within Microsoft's co-sell motions and customer search tools such as Microsoft AppSource.

For Security, the available specializations include the following:

- **Cloud Security**: Demonstrates your ability to secure hybrid and multicloud environments using Azure-native capabilities, including Microsoft Defender for Cloud
- **Identity and Access Management**: Focuses on your skill in managing identities securely using Microsoft Entra ID (formerly Azure AD), Conditional Access, and role-based access controls

- **Information Protection and Governance**: Highlights your competence in helping customers safeguard and govern sensitive data through Microsoft Purview, Data Loss Prevention (DLP), and related technologies
- **Threat Protection**: Recognizes expertise in deploying Microsoft Defender XDR or Sentinel to protect organizations against sophisticated threats

Requirements to qualify

To earn any of these specializations, partners must meet the following requirements:

- Hold an active Solutions Partner for Security designation
- Meet technical skilling requirements (specific certifications for each specialization)
- Show customer success, such as usage growth or active users
- Submit customer references or undergo an audit (depending on specialization)

Each specialization is valid for **one year**, with requalification typically requiring updated performance data and fresh customer validation.

For instance, for the *Threat Protection* specialization, you may need to demonstrate the following:

- Growth in Sentinel usage or Defender XDR deployment
- SC-200 or MS-500 certifications within your team
- At least one customer reference and evidence of workload success



Check out this link to access Microsoft's official decks explaining what the Solutions Partner for Security designation and technical requirements are: `https://partner.microsoft.com/en-rs/asset/collection/solutions-partner-for-security#/`.

With this foundation in place, the next section will focus on exploiting the partner programs strategically. You'll learn how to turn designations, specializations, and benefits into a concrete pipeline, funded engagements, accelerated solution development, and a differentiated security offering that stands out in a crowded Microsoft partner ecosystem.

Exploring security-focused partner programs

Microsoft offers a range of partner programs tailored specifically for security providers. These programs are designed to accelerate growth, enhance technical capabilities, and position your business as a trusted advisor within Microsoft's global customer and sales ecosystem. From marketing incentives to early access to new technologies, these programs help you differentiate and scale your security practice effectively.

Let's take a closer look at one of the most high-impact opportunities: the Microsoft-certified MXDR program.

Microsoft-certified MXDR partner program

The Microsoft-certified **Managed Extended Detection and Response (MXDR)** program recognizes partners who deliver end-to-end 24/7 managed security operations using Microsoft's security stack. These certified partners are trusted to provide expert threat hunting, custom detection engineering, real-time incident response, and comprehensive remediation, all through Microsoft's XDR technologies, such as Microsoft Defender and Sentinel.

Why should a security partner become a Microsoft-certified MXDR provider?

Becoming an MXDR-certified partner offers strategic advantages that can elevate your position in the market with the following benefits:

- **Integration via Partner APIs**: Seamlessly connect with Microsoft's threat intelligence and product ecosystem using Partner APIs to enhance threat detection and resolution workflows
- **Joint marketing initiatives**: Microsoft promotes your services through its commercial marketplace and field sales teams, significantly increasing your visibility
- **Sales enablement**: Gain access to exclusive tools and co-sell opportunities that help you engage customers faster and more effectively

In short, becoming a Microsoft-certified MXDR provider turns your service into an extension of the Microsoft security ecosystem rather than "*just another MSSP.*" You gain deep technical integration through Partner APIs, higher market visibility via Microsoft-led marketing and marketplace presence, and stronger sales momentum through co-sell motions and enablement tools. Together, this means more qualified opportunities, stickier customer relationships, and a defensible competitive edge built on Microsoft's own security platform.

Qualification process for Microsoft-certified MXDR solutions

Achieving Microsoft certification for your MXDR solution involves a comprehensive assessment and verification procedure:

- **Prerequisites compliance**: Initiating the process requires participation in Security Build Intent workshops and listing your MXDR solution on the Microsoft Commercial Marketplace. Consult your partner development team at Microsoft about further eligibility criteria.
- **Submit for validation**: Upon meeting the prerequisites, collaborate with your Microsoft partner development team to undergo your validation interview and submit your application.
- **Engineering verification**: Microsoft's engineering team evaluates your MXDR service delivery, solution architecture, and operational maturity.
- **Ongoing compliance**: Certified MXDR providers must undergo yearly reviews to ensure continued alignment with Microsoft's evolving standards.

In essence, the Microsoft-certified MXDR qualification process is designed to prove that your service is real, robust, and tightly aligned to Microsoft's standards. By meeting the prerequisites (workshops and marketplace listing), passing engineering-led verification of your architecture and operations, and maintaining yearly compliance checks, you demonstrate to customers and Microsoft alike that your MXDR is repeatable, resilient, and built on best practices. This rigor turns the certification into a powerful trust signal: it reassures buyers that your service has been technically validated and is being continuously held to a high operational bar.

Requirements for verified MXDR solutions

Becoming a verified MXDR partner requires that you, as a partner, meet the following criteria:

- **Established market presence**: Your MXDR offering should be actively promoted on your website and listed in the Microsoft Commercial Marketplace
- **Active security engagement**: Participation in Security Build Intent workshops within the past six months is essential
- **Validated membership**: Your organization must hold a Solutions Partner for Security designation and be part of the **Microsoft Intelligent Security Association (MISA)**
- **Customer endorsements**: Provide a minimum of three customer case studies highlighting your MXDR service success

Technical standards for verification

In terms of technical requirements for the verification to be successful, you must offer the following:

- **24/7/365 security operations**: Your service must deliver round-the-clock monitoring, response, and advanced hunting using Microsoft Defender XDR and Microsoft Sentinel
- **Security onboarding and configuration**: Capabilities must include deployment and configuration of Microsoft security solutions, with documented processes for tuning, baselining, and incident handling
- **Comprehensive incident life cycle management**: Your offering should include detection tuning, custom rule creation, threat analytics, incident triage, and both manual and automated response options
- **Managed Microsoft Sentinel services**: Demonstrate full-spectrum support for Microsoft Sentinel—from onboarding and configuration to proactive monitoring and incident response



Want to learn more?

Follow this link to discover the most recent requirements for joining the program: `https://microsoftpartners.microsoft.com/Microsoft-Security-Partners/`.

Taken together, the Microsoft-certified MXDR program and its verification requirements do more than add a logo to your website. They formalize your MXDR practice as a tested, trusted extension of the Microsoft security platform. By meeting technical, operational, and evidence-based criteria, you signal to customers and Microsoft that your service is always-on, outcome-focused, and built on repeatable best practices rather than ad hoc efforts. For security partners willing to invest, this becomes a powerful accelerator: it strengthens credibility, deepens alignment with Microsoft's roadmap, and creates a clear competitive edge in a crowded managed security market. Next, let's dive into another essential security program for partners.

Microsoft Intelligent Security Association

MISA is a curated, invite-only community of trusted Microsoft partners committed to advancing cybersecurity through collaboration and innovation. MISA brings together a select group of ISVs and **managed security service providers (MSSPs)** whose solutions deeply integrate with Microsoft's security stack, including Microsoft Sentinel, Defender XDR, Entra, and Purview. This global alliance is designed to foster a connected cybersecurity ecosystem where partners work alongside Microsoft to develop interoperable, intelligence-driven security solutions that strengthen defenses for customers worldwide.

MISA members contribute to and benefit from a collaborative security ecosystem that delivers value in several key areas:

- **Deeper integrations**: Members build solutions that align with Microsoft's architecture and APIs, enabling seamless interoperability across cloud, identity, compliance, and endpoint protection workloads
- **Elevated visibility and trust**: Being recognized as a MISA member signals to customers that your solutions meet Microsoft's high standards for security integration, scalability, and innovation
- **Customer-centric outcomes**: MISA helps partners co-innovate on solutions that deliver stronger, more cohesive protection against evolving threats, driven by unified signals, automation, and analytics
- **Go-to-market and enablement benefits**: Members gain exclusive access to joint marketing and co-branding opportunities, product roadmap previews, priority access to security events and partner roundtables, and networking opportunities with other elite security partners.

MISA acts as a quality stamp and force multiplier for security partners who are serious about building on the Microsoft stack. Membership signals deep technical alignment, unlocks co-innovation and go-to-market benefits, and embeds your solutions into a broader, intelligence-driven ecosystem. For customers, a MISA partner means tighter integrations, stronger outcomes, and solutions that are continuously shaped by Microsoft's security roadmap and community insight.

MISA membership pathway

MISA membership is not open enrollment but is by invitation only. Microsoft carefully vets prospective members based on their alignment with Microsoft's security platform and overall partner maturity.

ISVs must have validated integrations with Microsoft security products and demonstrate strong security solution value. **MSSPs** are assessed based on their delivery of Microsoft-powered managed detection and response services and Security Solutions Partner designation.

Once invited, partners go through a formal onboarding and technical validation process to ensure quality and readiness for co-marketing and solution showcase.



Explore Microsoft's official MISA program details at `https://aka.ms/misa`.

The MISA membership pathway is designed to keep the bar high. Rather than open enrollment, Microsoft handpicks ISVs and MSSPs that already demonstrate strong alignment with its security stack, proven integrations, and mature service delivery. The subsequent onboarding and technical validation ensure that only ready, credible partners make it into the program, so when customers see the MISA badge, they can trust that the partner has been thoroughly vetted for technical quality and joint go-to-market readiness. The next section is going to expand on a pivotal program called Cybersecurity Investments.

Microsoft Cybersecurity Investment Program

The **Cybersecurity Investment (CSI) Program** is a strategic Microsoft initiative designed to accelerate customer adoption of Microsoft security solutions through partner-led engagements (unified SecOps, threat protection, or data protection). Introduced as the evolution of Microsoft's earlier MSSP incentives model, CSI places certified security partners at the center of hands-on, solution-driven workshops aimed at boosting customer readiness, trust, and deployment.

CSI empowers Microsoft partners to deliver highly structured, value-focused engagements, which typically span 3 to 4 weeks, at no cost to eligible customers. These engagements help customers assess their current cybersecurity maturity, explore Microsoft's security stack, and build actionable roadmaps for implementation.

Each engagement is funded by Microsoft, which means that the partner is financially compensated for facilitating workshops that demonstrate the tangible value of Microsoft's security product.

Partner benefits and business impact

For partners, CSI is more than just a funding mechanism; it's a sales catalyst. Here's how it helps:

- **Direct funding from Microsoft**: Partners are compensated for each qualified engagement delivered, offering a predictable incentive model tied to tangible services
- **Pre-sales enablement**: Workshops are structured to address customer objections, evaluate competitive offerings, and position Microsoft solutions effectively
- **Accelerated pipeline conversion**: By demonstrating value in real environments, partners significantly boost win rates and shorten the sales cycle
- **Strategic alignment with Microsoft field teams**: Participation in CSI enhances visibility within Microsoft's go-to-market engine, improving co-sell collaboration and partner prioritization



Lessons from the field

Partners who consistently participate in the CSI program report stronger engagement with Microsoft account teams and greater traction in enterprise sales motions. Since these workshops lower the barrier to entry for customers, they're especially useful when prospects are evaluating multiple vendors or are unsure which Microsoft product to adopt.

In short, CSI is a win-win model: customers receive expert-led assessments at no cost, and partners are rewarded for driving security adoption in a consultative, solution-oriented way.

Learn more about the available workshops here: `https://cloudaccelerator.microsoft.com/csi-partner-investment`.

Understanding the relationship between CSI and MCI

As a Microsoft security partner, it's important to recognize the distinction between two frequently referenced programs: the CSI program and the **Microsoft Commerce Incentives** (**MCI**) program. Although they are interconnected, each serves a different purpose within Microsoft's partner ecosystem.

The MCI program serves as Microsoft's broader incentives framework. It governs how partners are financially rewarded for delivering services, driving product usage, and influencing cloud adoption across solution areas, including Azure, Microsoft 365, Security, and Dynamics 365. MCI enables partners to earn rebates, co-op marketing funds, and usage-based incentives. To participate, partners must be enrolled in the MCI program via Partner Center and meet specific eligibility requirements tied to Microsoft solution areas.

While CSI and MCI are distinct, they are closely linked operationally. CSI-funded engagements are tracked and managed through the MCI framework. This means that partners submit CSI workshop claims through the MCI platform, Microsoft reviews and approves incentive payments via MCI, and all CSI engagements must align with MCI requirements and reporting protocols. In other words, MCI is the financial infrastructure that powers CSI and other funded partner engagements.

Let's now explore a fundamental engagement within the MCI program: the Microsoft Sentinel Migration and Modernization Program.

Microsoft Sentinel Migration and Modernization Program

The Microsoft Sentinel **Migration and Modernization Program (MMP)** is a partner-led initiative designed to help organizations transition from traditional or legacy SIEM platforms to Microsoft Sentinel, Microsoft's scalable, cloud-native SIEM and SOAR solution. This program is delivered exclusively through certified Microsoft partners who possess deep expertise in Sentinel deployments and threat protection.

MMP supports organizations not only in moving their existing operations but also in modernizing their security infrastructure by leveraging Sentinel's full potential, including its native integration with Microsoft Defender XDR and Entra ID, AI-powered analytics, and flexible architecture.

Core components of the program

The following are the key components of MMP:

- **Migration support**: Partners guide customers through a seamless transition from legacy SIEM tools such as Splunk, ArcSight, or QRadar to Microsoft Sentinel. This includes migration strategy planning, data mapping, log ingestion setup, configuration of analytics rules, and preservation of existing SOC workflows.
- **Security operations modernization**: MMP is more than a "lift and shift." Partners help customers take advantage of Sentinel's advanced capabilities, such as automation via Logic Apps, integration with threat intelligence feeds, customized detection engineering, and cloud-optimized scalability.

- **Partner-driven delivery**: All engagements are led by Microsoft-approved partners with proven Sentinel and security expertise. These partners offer end-to-end services, from assessment and architecture design to deployment, tuning, and SOC training, ensuring a smooth adoption journey.
- **Program benefits**: These include reduced complexity and faster time-to-value for customers, access to Microsoft-funded engagements that lower upfront costs, minimized downtime during SIEM transition, and improved threat visibility, response automation, and overall SOC maturity.
- **Customer eligibility**: The program is available to enterprise and SMC-Corporate select customers who have an active agreement with Microsoft and a recognized Top Parent ID within Microsoft systems. Funding eligibility is determined by Microsoft's internal validation.



Lessons from the field

To tailor support effectively, Microsoft classifies Sentinel migration projects into three categories based on projected annual Sentinel consumption:

- Small: $25,000–$125,000
- Medium: $125,000–$250,000
- Large: $250,000–$500,000

This tiered approach ensures that partner engagements are appropriately scoped and resourced, aligning technical complexity and customer expectations.

Budget constraints often pose a major barrier to SIEM modernization. This program helps alleviate that concern by providing Microsoft-funded support, allowing customers to begin their cloud SIEM journey without shouldering the full financial burden.

Find out more about this program by following this link: `https://partner.microsoft.com/en-us/partnership/partner-incentives/microsoft-sentinel-migration-and-modernization`.

Microsoft offers a rich ecosystem of programs that can help you accelerate customer adoption. By leveraging these options, you can reduce time-to-value for new customers, access structured support from Microsoft, and build a more predictable, scalable delivery model. With that foundation in place, the next step is to look at how Microsoft FastTrack can further streamline onboarding and deployment, helping you move from opportunity to successful, production-grade implementations even faster.

Microsoft FastTrack partners

Microsoft FastTrack is a customer success program designed to help organizations adopt Microsoft cloud technologies with speed, confidence, and minimal disruption. Delivered in collaboration with approved partners, FastTrack is a key part of Microsoft's strategy to accelerate cloud transformation for customers across industries.

FastTrack partners are a select group of Microsoft-certified service providers who have demonstrated deep expertise in deploying Microsoft cloud solutions. These include Microsoft 365, Azure, Dynamics 365, and Microsoft security platforms. Recognized for their delivery excellence, FastTrack partners work closely with Microsoft engineering and customer success teams to provide high-touch onboarding and migration support. Their role is to help organizations plan, implement, and optimize cloud deployments, ensuring that customers not only go live but also realize meaningful value from their investment.

What do FastTrack partners deliver?

FastTrack partners support customers across all phases of their cloud journey, including the following:

- **Migration and deployment planning**: Develop tailored rollout plans aligned to business needs and technical requirements
- **Cloud service onboarding**: Assist with the configuration, provisioning, and foundational setup of Microsoft cloud environments
- **User adoption and change management**: Provide training, resources, and best practices to ensure that users embrace new tools and workflows
- **Ongoing optimization**: Help organizations continuously improve their environments, reduce risk, and drive long-term success with Microsoft solutions

These partners also have access to proprietary Microsoft tools, methodologies, and technical guidance to ensure best-practice delivery.



FastTrack services are typically available at no extra cost to customers with eligible Microsoft subscriptions. By working with a FastTrack partner, organizations can streamline their move to the cloud, accelerate time-to-value, and reduce common deployment roadblocks, all while leveraging expert support backed by Microsoft.

Partner security: protecting your customers

Now that we've examined Microsoft's security-focused partner programs, it's time to focus on your role as a security partner and the responsibility that comes with it. Whether you're operating as a **cloud solution provider** (**CSP**), MSSP, or any other Microsoft-certified security partner, you hold privileged access to customer environments. With this trust comes significant responsibility.

Your ability to safeguard your customers is not only a cornerstone of your credibility but also a contractual and operational obligation. Microsoft grants partners elevated access rights to administer services, but also expects them to meet strict security standards to ensure that trust isn't compromised.

In the following section, we will explore essential security controls that are crucial for maintaining a secure environment for both you and your partners.

Mandatory security requirements for Microsoft partners

As part of the Microsoft Partner Agreement, all participating partners must adhere to a set of evolving security controls designed to minimize risk, protect customer data, and preserve the integrity of the CSP ecosystem. Two core requirements are MFA and a secure application model.

Enforced across all partner accounts, MFA is mandatory for accessing Microsoft cloud services and for transactions managed through Partner Center. Microsoft accepts both its native MFA solution and approved third-party alternatives. Failure to implement MFA by the defined deadlines may result in loss of access or suspension of partner privileges. With a secure application model, partners are expected to authenticate apps securely, minimizing risk associated with overly broad application permissions.

Microsoft updates these requirements periodically, and non-compliance can result in escalated enforcement, including revoked access rights or reduced Partner Center capabilities. As such, staying ahead of these obligations is essential for maintaining eligibility and customer trust.

Customer access models: from DAP to GDAP

Managing customer environments requires controlled access. Microsoft provides two frameworks to do this: **Delegated Admin Privileges** (**DAP**) and the newer, more secure **Granular Delegated Admin Privileges** (**GDAP**).

DAP

DAP enables partners to manage their customers' services or subscriptions with explicit permission for admin access. To ensure the security of customer accounts and assets in Partner Center, Microsoft recommends that partners terminate any DAP that are no longer in use. Moreover, Microsoft is transitioning away from DAP in favor of a more secure, restricted access model known as GDAP. This new framework offers enhanced security by allowing partners to have more precise control over access to customer services, tailored specifically to the needs of each customer's environment.

GDAP

GDAP is Microsoft's modern approach to partner-customer access, rooted in the principles of **Zero Trust**. Unlike DAP, which grants broad and persistent access, GDAP enables **fine-grained, time-limited, and explicitly consented access** to specific roles (M365 built-in roles) and services within a customer tenant.

Key features of GDAP include the following:

- **Role-based access control** (**RBAC**): Assign only the permissions needed, nothing more
- **Time-bound permissions**: Set expiration dates to reduce persistent access risk
- **Explicit customer consent**: Customers approve exact roles and scopes before access is granted
- **Auditability**: All GDAP activities are logged and can be reviewed for compliance and oversight

By aligning with GDAP, partners significantly reduce exposure to misuse or compromise, while maintaining secure, least-privilege access to customer environments.



For partners, embracing GDAP is quickly becoming an operational necessity rather than just a best practice. Microsoft provides dedicated tools and guidance through Partner Center to streamline GDAP setup, access request workflows, and life cycle management.

Figure 3.2 provides an architectural overview of GDAP:

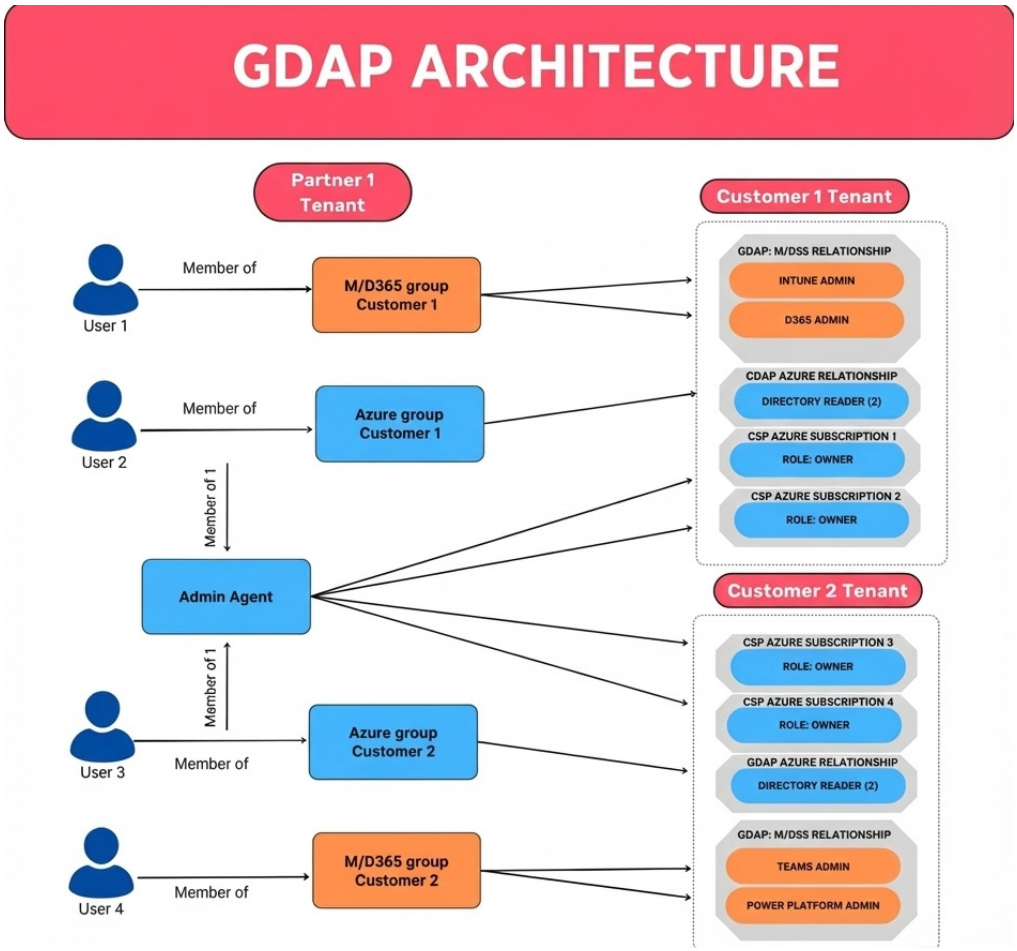


Figure 3.2: GDAP architecture



Update: Reduced default permissions for Azure management:

Microsoft has updated access protocols for partners managing Azure environments. Partners no longer receive **Global Administrator** privileges by default when working within a customer's tenant. Instead, they are granted **restricted, read-only directory access**, significantly minimizing potential risk in line with Microsoft's least privilege approach.

DAP to GDAP transition has no impact on Partner Earned Credit (PEC):

Transitioning from DAP to GDAP does not affect your PEC. Partners can confidently remove legacy DAP permissions, including Global Admin access, without compromising the recognition or incentives tied to customer impact.

Accessing customers' tenants securely is important as it reduces the attack surface and helps keep your customers safe.

Now, let's look at how you can set up notifications to detect fraud within your customers' Azure environment.

Azure fraud notifications

Within the CSP program, it's vital to oversee your clients' Azure usage, paying particular attention to unauthorized cryptocurrency mining activities. Promptly spotting any irregular behavior facilitates immediate checks on its legitimacy, and if necessary, the quick suspension of affected Azure resources or subscriptions to mitigate any risks.

Based on field experience, **Microsoft is unlikely to waive charges** associated with unauthorized Azure usage, especially in cases involving crypto mining. If a customer's tenant is compromised and threat actors deploy compute-intensive virtual machines, the resulting costs typically fall on the customer.

While Microsoft may assess situations on a case-by-case basis, relying on retrospective appeals isn't a reliable risk mitigation strategy. Proactive monitoring and secure configurations are your best defense.



Strengthening detection with Microsoft Entra ID P2

To enhance your CSP security posture, consider upgrading to Microsoft Entra ID P2. This license offers advanced identity protection features, including the following:

- **Privileged identity management (PIM)**: Enforce just-in-time access for admin roles
- Sign-in logs and risk-based Conditional Access: Gain insights into potentially malicious logins and respond automatically
- Granular role monitoring: Track and review who is accessing what, and when

Up next, let's explore some quick wins and high-impact security improvements that every partner can implement to enhance protection across customer environments.

Get recognized by Microsoft for your success: CPOR and PAL

As a Microsoft security partner, driving customer adoption and delivering value is only part of the journey. To truly benefit from your hard work, you need to make sure Microsoft recognizes your influence. Two key mechanisms exist for this purpose: CPOR and PAL.

These tools help Microsoft track your contribution to customer success, reward your influence through incentive programs, and ensure that you gain visibility for the impact you're making in the field.

CPOR

CPOR is Microsoft's official model for associating partners with customer workloads in Microsoft 365 and other cloud services. It allows partners to be recognized for driving product adoption, influencing purchasing decisions, or delivering services that lead to successful deployments.

With CPOR in place, partners gain access to customer usage telemetry, helping them monitor product consumption trends, identify opportunities to increase value, and guide customers through adoption planning. Importantly, CPOR grants visibility into **usage data only.** No personal or sensitive customer information is shared. This ensures both insight and privacy are maintained.

CPOR plays a key role in helping Microsoft by accurately recognizing the partners responsible for successful implementations, by rewarding partners through **usage-based incentives** such as Modern Work & Security Usage incentives, and **Online Services Advisor** (**OSA**), and by allocating field resources and sales support more effectively based on partner performance. Unlike the CSP relationship, which is transactional, CPOR is about **influence**. It ties your expertise and value delivery directly to specific customer workloads, often where you're not transacting the license but are driving outcomes.

How to claim CPOR

To claim a customer workload under CPOR, partners must meet the following requirements:

1. Submit **proof of execution** through Partner Center.
2. Validate their influence on the workload (e.g., through customer confirmation or usage metrics).
3. Respond to any verification requests, if required.

If a claim is denied, partners can challenge it or withdraw from the association at any time.



Follow this link to download the latest CPOR association guide by Microsoft. It explains incentives, usage, revenue, and the claim process:

`https://partner.microsoft.com/en-us/asset/collection/claiming-partner-of-record-cpor-resources#/ Partner Admin Link`.

Up next, we'll dive into PAL, which is another key method for ensuring that Microsoft recognizes your work, especially when operating in Azure environments.

PAL: get recognized for Azure impact

PAL is one of Microsoft's most important tools for partners looking to gain recognition for their work in Azure environments. Designed specifically to support managed services scenarios, PAL enables you to associate your **Microsoft Partner Network** (**MPN**) ID with your user identity when managing customer resources in Azure.

This association allows Microsoft to track and credit your contribution to a customer's Azure usage, even when the customer is the one incurring the cost. Whether you're deploying Microsoft Sentinel, configuring security controls, or delivering ongoing operational support, PAL ensures your efforts are visible to Microsoft.

PAL doesn't grant access or permissions. Instead, it simply **tags your identity** (user, service principal, or Lighthouse delegation) with your **MPN ID**. This creates a link between your actions and the Azure resources you're managing. Microsoft can then measure your influence on ACR and apply it toward your partner performance metrics and incentives.

Key benefits of PAL include visibility and recognition for your Azure work, eligibility for usage-based incentives, improved partner telemetry and qualification for managed partner status, and attribution of security solution impact, particularly through Microsoft Sentinel. Most importantly, PAL is separate from CSP billing relationships and has no impact on access levels or service control. It's purely a reporting mechanism to ensure your value doesn't go unnoticed.



Lessons from the field

For partners delivering managed security services, especially through Microsoft Sentinel, PAL is the most effective way to capture credit for customer Azure spend. While your customer pays for the Azure resources, your engineering work, which includes deploying and managing those resources, is what drives usage. PAL bridges that gap and ensures that your contribution is counted.

If your goal is to become a **Managed Partner** or build a case for co-sell alignment, PAL can help you demonstrate customer reach, workload impact, and revenue influence, which are the key metrics Microsoft uses to prioritize partner engagement.

Getting PAL set up

There are several ways to establish PAL, depending on how you access your customer's Azure environment. One of the key elements to achieve PAL recognition is to be eligible for PEC, as you must meet the following criteria:

- Be an active member of the Microsoft AI Cloud Partner Program
- Hold a valid RBAC role on the resource, resource group, or subscription level
- Use one of the following access methods: **Admin-on-Behalf-Of** (**AOBO**), direct guest access (B2B account), Azure Lighthouse, or service principal

PEC is earned on the Azure resource level, resource group, or subscription. If a partner has valid access at either the subscription or resource group level, each resource that rolls up to the higher entity earns PEC.

PAL setup

The simplest and most scalable way to configure PAL is by deploying an **Azure Lighthouse** delegation. When you onboard a customer tenant via Lighthouse and assign a qualifying RBAC role, your partner tenant is automatically linked, and PAL is established behind the scenes.

Other setup options include direct guest invites, B2B accounts, or service principals, though these require additional configuration steps. Suitable roles that follow the principle of least privilege and are also eligible for PEC, which we recommend, include the following:

Role name	Description	PEC eligible
Support Request Contributor	Let's you create and manage support requests	Yes
Cost Management Contributor	Can view costs and manage cost configuration (for example, budgets, exports)	Yes

Table 3.2: PAL active suggested roles

A full list of roles can be found here: `https://learn.microsoft.com/en-us/partner-center/azure-roles-perms-pec`.



Lessons from the field

Before setting up PAL, take the time to understand PEC. PEC is Microsoft's mechanism for rewarding partners who own the operational control of a customer's Azure workloads. If PAL is how you're recognized, PEC is how you get rewarded.

Azure Lighthouse deployment

Once you have access to the client's environment, or once the customer is ready to proceed with the next steps, you can accomplish this in several ways; we chose a template as an effective demonstration to highlight the roles.

You will first need to gather the following details:

- **Service provider tenant ID**: This is your tenant ID, the environment from which you'll manage the customer's Azure resources.
- **Customer tenant ID**: The tenant ID of the customer whose subscriptions or resource groups you'll be managing.
- **Customer subscription ID(s)**: The specific subscription(s) where delegated access will be granted. You can also scope this further into specific **resource groups** within a subscription, if needed.
- **Security group or service principal ID**: You'll need to specify the object ID of the **Entra ID security group** or **service principal** within your tenant that will be granted access through the delegation. This ensures that the correct users or automation identities are allowed to manage the customer environment.



You can deploy the Azure Lighthouse ARM template directly through the Azure portal, Azure CLI, or PowerShell. Microsoft also offers official deployment templates via GitHub, which you can customize for your environment and required roles. Here's the link: `https://github.com/Azure/Azure-Lighthouse-samples`.

Whatever method you decide to utilize, ensure that it is done properly, as this is how Microsoft and you, as a partner, can track the revenue generated by your security services. Again, we emphasize that setting up PAL using an Azure Lighthouse connection is the most effective and hassle-free method. Don't forget to also get your PEC role assigned.

Summary

This chapter provided a comprehensive overview of the Microsoft Partner Center and the key programs and designations available to Microsoft security partners. It began by exploring the Microsoft AI Cloud Partner Program, followed by a breakdown of the Solutions Partner designation, which showcases an organization's capabilities across areas such as Business Applications, Data & AI, Infrastructure, Modern Work, and Security.

We then explored several security-focused initiatives, including the Microsoft-certified MXDR program for managed detection and response, MISA for integrated security vendors, the CSI program for Microsoft-funded workshops, and the Sentinel Migration and Modernization program to help customers transition to Microsoft's cloud-native SIEM.

The second part of the chapter shifted focus to essential partner security practices for protecting customer environments. We covered Microsoft's mandatory security requirements, such as MFA, and recommended best practices such as GDAP. We also outlined a series of quick wins for enhancing partner and customer security posture.

Finally, we explored how partners can gain formal recognition from Microsoft for the value they deliver. We discussed CPOR, which attributes Microsoft 365 workload influence on partners, and PAL, which ties Azure service delivery to a partner's MPN ID, enabling Microsoft to credit Azure growth to the appropriate security partner.

In the next chapter, we'll dive into the core foundations of Microsoft XDR, focusing on the essential components and considerations for building a strong, secure architecture with Microsoft security solutions.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW

4

Core Foundations: Initial Setup, Strategic Financial Planning, and Business Insights

This chapter lays the groundwork for building a secure, scalable, and cost-effective Microsoft Security deployment, with a focus on Microsoft Sentinel as the central SIEM platform. Whether you're establishing your first Microsoft 365 tenant or scaling a multitenant security operations model, the success of your deployment starts with strong architectural decisions, effective governance, and a clear understanding of financial impact.

In this chapter, we'll cover the following topics:

- Foundational requirements: Setting up Microsoft 365 tenants and Azure with governance and security baselines
- Choosing your workspace architecture: Evaluating centralized, decentralized, and hybrid options
- Designing **role-based access control** (**RBAC**) for Microsoft Sentinel
- Sentinel pre-deployment checklist: Planning with precision
- Multitenant management with Defender XDR and Microsoft MTO
- Deployment roadmaps: Migration and modernization of your SIEM solution
- Data retention and long-term storage
- Financial journey: Assessing costs from groundwork to full operation

Let's begin by establishing your foundational environment with Microsoft 365 and Azure, which sets the stage for everything that follows.

Foundational requirements: Setting up Microsoft 365 tenants and Azure

Before diving into the advanced capabilities of Microsoft's security suite, it's important to establish a secure and well-structured foundation. This section will help you set up Microsoft 365 tenants and Azure subscriptions, which form the backbone of any Microsoft Sentinel deployment.

What is a Microsoft 365 tenant?

A Microsoft 365 tenant is a dedicated, cloud-based instance of Microsoft's productivity and collaboration services, provisioned exclusively for your organization. It forms the administrative and security boundary for managing users, data, apps, and policies.

When an organization subscribes to Microsoft 365, a unique tenant with its own domain and directory is assigned, which is isolated from other organizations in the Microsoft cloud. This tenant enables centralized control over services such as Exchange Online, SharePoint Online, Teams, and Entra ID, while ensuring data integrity and access control.

Each tenant provides logical and security isolation from other organizations, a single pane of glass for **identity and access management (IAM)**, and the governance structure for assigning roles, licensing, and service policies. Understanding and configuring your tenant correctly is the first step toward building a secure, compliant, and scalable cloud environment.

Table 4.1 outlines the attributes of a well-structured Microsoft 365 tenant:

Area	Focus	Best Practice
Networking	Network configuration and optimization	You have configured the correct DNS domain names. For enterprise networks, you have optimized network traffic to the Microsoft network for on-site workers. You have optimized network traffic for remote workers who are using a VPN client.

Area	Focus	Best Practice
Active Directory Domain Services	Directory synchronization and identity alignment	You have synchronized your Active Directory Domain Services (AD DS) accounts, groups, and other objects. Your Microsoft Entra tenant accounts are mapped to Exchange Online mailboxes with the correct DNS domains for email addresses. Your user accounts have been assigned the correct licenses from the correct purchased products, such as Microsoft 365 E3 or E5.
Identity and access management	Authentication and access control	You have configured strong IAM. You are requiring secure user sign-in with passwordless or multi-factor authentication (MFA). Conditional Access policies are enforced to apply sign-in requirements and restrictions for higher levels of security.
Device management	Endpoint and application management	You are managing devices with Intune or Basic Mobility and Security built into Microsoft 365. Your organization-owned devices are enrolled and managed. Applications on personal devices are managed using appropriate app management controls.

Table 4.1: Attributes of a well-designed Microsoft 365 tenant

Let's now look at a few quick wins that can help establish a strong security baseline right from the start for your Microsoft 365 tenant.

Configuring your Microsoft 365 tenant for enhanced security

While setting up a Microsoft 365 tenant is relatively straightforward, its simplicity can often lead to critical security measures being overlooked. This section will give you practical guidance on strengthening your tenant's security posture from day one. Following this, ensure that your environment is not just operational but also resilient against modern threats.

Start with the fundamentals

One of the most exploited attack vectors in cloud environments is email. As numerous breach reports confirm, email remains the primary entry point for phishing, malware, and credential theft. That's why securing this layer should be a top priority in your initial configuration.

To protect your organization against email attacks, Microsoft provides two key security offerings: **Exchange Online Protection** (**EOP**) and **Microsoft Defender for Office 365** (**MDO**):

- **EOP**: This is included with all Microsoft 365 tenants and provides foundational anti-spam, anti-malware, and policy enforcement capabilities. It's a good starting point for basic domain-level protection.
- **MDO**: This builds on EOP with advanced threat protection features such as Safe Links, Safe Attachments, real-time detections, and automated investigation and response. It is designed to defend against sophisticated, targeted attacks and is especially valuable for organizations at higher risk or operating in regulated industries.



Worth exploring

If your organization is ready to take email security further, MDO Plan 2 offers the full suite of protection tools, including threat analytics, attack simulation training, automated investigation and remediation, real-time detection, and insights across mail flow.

Microsoft offers a 90-day trial of Plan 2 at no cost, allowing you to evaluate the capabilities in a real-world context.

Visit `https://security.microsoft.com/trialHorizontalHub?sku=MDO&ref=DocsRef` to explore available trials and see what fits your organization's needs.

Once you've accessed the preceding URL, you will see the Microsoft Security trials portal as shown in *Figure 4.1*. Keep in mind that each trial can only be activated once.

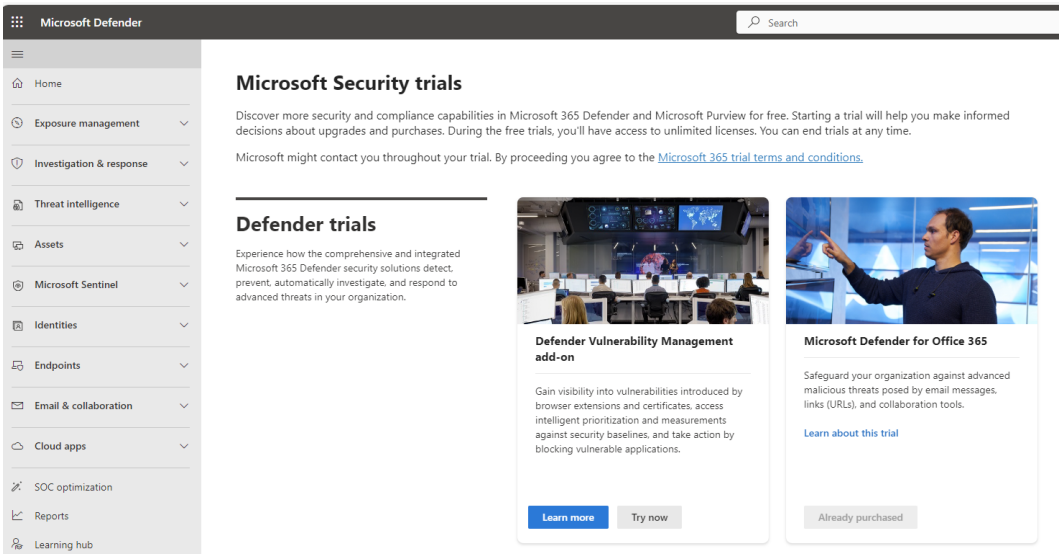


Figure 4.1: Microsoft Security trials

Strengthening and protecting your Microsoft 365 domain doesn't have to be complex. With a few essential actions, you can significantly enhance your organization's security posture, especially when it comes to email.

Let's look at some of these actions to secure your Microsoft 365 environment with minimal overhead, beginning with foundational email authentication controls.

Securing your email with authentication protocols

It is important to implement email authentication to defend against spoofing, phishing, and domain impersonation. These controls validate that emails sent from your domain are legitimate, protecting your brand and your recipients.

Focus on configuring these three key protocols for all domains in your tenant, including parked and subdomains:

- **Sender Policy Framework (SPF)**: This acts as your first layer of defense by specifying which mail servers are authorized to send emails on behalf of your domain. SPF works by publishing a record in your DNS.
- **DomainKeys Identified Mail (DKIM)**: Digitally signs your outbound emails using cryptographic keys, helping recipient servers verify the integrity of the message even if it's been forwarded.

- **Domain-based Message Authentication, Reporting, and Conformance (DMARC)**: Builds on SPF and DKIM by setting a policy for how receiving servers should handle unauthenticated messages. You can instruct servers to monitor, quarantine, or reject messages, providing full control over how your domain is protected.

Once these protocols are in place, the next step is to apply protection policies that will actively detect and stop malicious content.

Leveraging protection policies in EOP and MDO

EOP and MDO include built-in policy frameworks that enhance your email security posture. The following are your available options:

- **Default policies**: Automatically applied during tenant setup, these provide baseline protection. While not fully customizable, settings such as spam filtering thresholds and malware detection levels can be fine-tuned.
- **Preset security policies**: Microsoft offers preconfigured policies (Standard, Strict, and Built-in) to accommodate organizations at different security maturity levels. Start with the Standard preset for a balanced approach between protection and usability. As your team gains experience, you can transition to the Strict preset for more aggressive filtering.
- **Custom policies**: For organizations needing granular control, custom policies allow you to define conditions, exceptions, and layered defenses tailored to specific users, groups, or domains. These are ideal for adapting to your unique threat landscape.



For a detailed, step-by-step guide on configuring each setting, check out Microsoft's documentation on email authentication. This guide covers every step you need to make these configurations seamless and effective: `https://learn.microsoft.com/en-us/microsoft-365/security/office-365-security/email-authentication-about?view=o365-worldwide`.

Now that you've established strong email security using EOP and MDO, it's time to shift focus to another important pillar of cybersecurity: IAM.

Identity and access management (IAM)

IAM is one of the most impactful areas of security configuration in Microsoft 365, and where strong identity controls help prevent compromise and enforce a zero-trust strategy. Although we will briefly discuss IAM and securing identity in this chapter, the topic will be explored in greater depth in *Chapter 6*.

For now, let's focus on one of the simplest, most effective ways to enhance identity security across your Microsoft 365 environment.

Microsoft Entra ID security defaults

Security defaults in Microsoft Entra ID (formerly Azure Active Directory) provide out-of-the-box identity protection at no additional cost. These are designed to enforce best practices with minimal configuration and are especially helpful for small to mid-sized organizations without dedicated identity architects.

By default, these settings are automatically enabled for all new tenants created after October 22, 2019, unless overridden by premium configurations such as Conditional Access policies.

Core protections include the following:

- **Mandatory MFA**: All users must register for MFA. Admins must authenticate using MFA every time they sign in.
- **Conditional MFA prompts**: Rather than applying MFA universally, prompts are issued dynamically based on real-time risk signals and user behavior. This helps create a balance between security and user experience.
- **Blocking legacy authentication**: Protocols such as **Post Office Protocol (POP)**, **Internet Message Access Protocol (IMAP)**, and **Simple Mail Transfer Protocol Authentication (SMTP AUTH)**, which lack modern authentication, are disabled by default to reduce the attack surface.
- **Privileged access hardening**: High-impact activities, such as accessing the Microsoft Entra admin portal, trigger additional verification to protect privileged accounts from abuse.

Security defaults are recommended when you have no Conditional Access policies currently in place, the tenant is not using Microsoft Entra Premium features, or no legacy authentication clients are required for business apps. If your tenant meets these criteria, Microsoft will continue prompting administrators to enable security defaults until action is taken.

Now that you've taken the first step in understanding what security defaults offer, let's walk through how to implement them in your tenant.

Managing security defaults

Whether you're enabling the security defaults for baseline protection or disabling them in favor of more advanced Conditional Access policies, knowing where to find these settings is key. We will now discuss how to access and manage security defaults within the Microsoft Entra admin center.

To enable or disable security defaults in your Microsoft 365 tenant, follow these steps:

1. Sign in to the Microsoft Entra admin center (`https://entra.microsoft.com/`) using an account with Security Administrator privileges.
2. In the left-hand navigation pane, go to **Identity | Overview | Properties**.
3. Scroll to the **Manage security defaults** section and select **Enable** or **Disable** based on your organization's requirements.



If you're not using Conditional Access policies or premium features and your environment doesn't require legacy authentication, enabling security defaults is one of the easiest ways to level up your identity security instantly.

Expanding an IAM strategy beyond defaults

As your organization grows or faces more complex regulatory, operational, or collaboration needs, it's essential to move beyond the default settings and embrace a more mature IAM strategy.

Once you've established a secure foundation with security defaults, consider incorporating these advanced IAM capabilities for greater control, flexibility, and compliance:

- **Entra External ID management**: Securely collaborate with external users such as business partners, contractors, or vendors, using Azure B2B collaboration and guest access features. Enforce policies that safeguard internal resources while enabling seamless collaboration.
- **Privileged access management**: Limit the exposure of elevated roles with **just-in-time (JIT)** access and role elevation through **privileged identity management (PIM)**. This helps ensure that administrative privileges are only active when absolutely necessary.

  Remember, this doesn't supersede the need for separate admin/BAU accounts. It just adds to limiting exposure.

- **Identity governance**: Implement access reviews, entitlement management, and life cycle workflows to enforce least privilege, maintain compliance, and manage access as employees join, move, or leave the organization.

These features are part of the Microsoft Entra Premium P1 and P2 SKUs.

As identity and access capabilities evolve, so does the need to manage them across larger, more distributed environments. Whether your business has acquired subsidiaries, operates in multiple geographies, or simply wants to streamline collaboration between divisions, identity management across multiple Microsoft 365 tenants can become a challenge. That's where **Multitenant Organizations (MTOs)** come in.

Multitenant organizations in Microsoft 365

Now publicly available, MTOs in Microsoft 365 provide a modern, scalable way to unify identity and collaboration across multiple Microsoft tenants. This is a game-changer for large enterprises and conglomerates operating in complex or federated environments. For administrators, you get a more standardized approach to cross-tenant collaboration, anchored in Entra policy and life cycle automation (create/update/remove synced identities), rather than relying on ad hoc guest invites and inconsistent cross-tenant rules.

Microsoft currently has a service limit of up to 100 active tenants in an MTO (including the owner tenant), which is why this can work for complex, federated enterprise structures.

With MTOs, organizations can synchronize users across tenants to ensure seamless directory consistency. MTOs help organizations configure cross-tenant access settings, enabling trusted collaboration across business units. It also helps them share resources and data securely, breaking down the silos between separate tenants while maintaining governance. This is especially valuable for companies with multiple subsidiaries or global divisions that have historically operated under separate tenants but now require centralized visibility and coordination.



For a full walk-through on how to plan and deploy this feature, refer to Microsoft's official guidance here: `https://learn.microsoft.com/en-us/microsoft-365/enterprise/plan-multi-tenant-org-overview?view=o365-worldwide`.

With your identity strategy now taking shape, it's time to turn our attention to the infrastructure layer of your Microsoft cloud environment: Azure subscriptions.

Azure subscriptions

In Microsoft Security, Azure subscriptions enable organizations to deploy, manage, and scale cloud-based security solutions effectively. An Azure subscription acts as a logical container for Azure resources such as Microsoft Sentinel and establishes a framework for billing, access control, and resource management. It is worth explicitly calling out the subscription-to-tenant relationship because it is a hard boundary that drives governance, billing, identity, and operational design.

One Microsoft Entra ID tenant can contain many Azure subscriptions. Each Azure subscription is associated with exactly one tenant at any point in time (you cannot have a single subscription spanning multiple tenants).

Why does this matter in practice? If your security, logging, policy, or access model assumes "one tenant = one subscription" or treats subscriptions as portable between tenants without change control, you will create design gaps, particularly around RBAC, Conditional Access, Azure Policy inheritance, and centralized Sentinel onboarding.

This section will discuss Azure subscriptions and guide you to choose the right subscription model for your organization. You will also learn how to create and configure subscriptions through a step-by-step walk-through.

Understanding and choosing subscription models

Microsoft offers multiple subscription models, each tailored to different business sizes, usage patterns, and long-term goals:

- **Pay-as-you-go (PAYG)**: This subscription is ideal for smaller organizations or experimental deployments. You're billed based on actual consumption, giving you maximum flexibility.
- **Enterprise Agreement (EA)**: This is designed for large enterprises with predictable workloads. It offers custom pricing, consolidated billing, and advanced support for long-term cloud strategies.
- **Microsoft Customer Agreement (MCA)**: This subscription is a modern, simplified experience suitable for organizations of all sizes. MCA provides flexible payment options, consolidated billing, and strong cost management features.



Which model is right for you?

Your choice should reflect your organization's size, budget, and cloud adoption strategy. For deeper guidance, visit Microsoft's official overview documentation: `https://learn.microsoft.com/en-us/azure/cost-management-billing/manage/`.

Azure subscription best practices

Each Azure subscription serves as a container for cloud resources, enabling access control, resource management, and billing. By structuring subscriptions hierarchically, which means using management groups, subscriptions, resource groups, and individual resources, you can achieve precise governance and control across your Azure environment.

To get started with Azure security configuration and initial setup, revisit *Chapter 2* for insights on the Cloud Adoption Framework and the Well-Architected Framework for Azure, where we've discussed foundational best practices for building a robust cloud strategy. We recommend that you consider those practices when you're planning to set up a new Azure subscription.

Structuring your Azure subscriptions

As your environment grows, you need a well-structured Azure subscription to manage costs efficiently, enforce security boundaries, and enable support. The following are some practical approaches that can be used as a starting point for organizing subscriptions and resources:

- **Billing and cost management**: Separate subscriptions by department, team, or project to track costs accurately and assign budgets effectively.
- **Start simple**: If you're just beginning, a single subscription can simplify your setup. As needs evolve, introduce new subscriptions incrementally.
- **Access control and security boundaries**: Use subscriptions to enforce least privilege and isolate risk. Assign roles carefully to control access per team or project.
- **Compliance and regulatory segregation**: Create dedicated subscriptions for workloads governed by specific regulations (e.g., HIPAA or GDPR) to maintain compliance boundaries
- **Separate production and non-production**: Distinguish between environments (e.g., production, development, and test) to prevent accidental changes and maintain control.
- **Leverage management groups**: Use management groups to apply policies and compliance rules across multiple subscriptions. This is ideal for enterprise environments.
- **Use resource groups for logical grouping**: Organize assets within each subscription by resource group, aligning with the project life cycle, function, or team structure.
- **Consistent naming conventions**: Adopt standardized naming across subscriptions and resources to support clarity, automation, and policy enforcement.
- **Plan for scalability and flexibility**: Think ahead and assume that your Azure footprint will grow. A well-architected subscription model supports future expansion without rework.



Go back to the discussion we had about the Well-Architected Framework and the Cloud Adoption Framework for Azure in *Chapter 2*. These will give you strong foundations to build your subscription in a secure and scalable manner.

Once your Azure subscription model is defined, the next decision to make in your Microsoft Security journey is how to structure your Log Analytics workspaces for Microsoft Sentinel.

Choosing your workspace architecture for Microsoft Sentinel: Centralized, decentralized, or hybrid

One of the early and most strategic decisions in any Microsoft Sentinel deployment is how to architect your Log Analytics workspaces. Microsoft supports three primary models: centralized, decentralized, and hybrid (a blend of the two). Each has its own operational trade-offs.

We'll begin by discussing the centralized workspace architecture, where all telemetry and security logs are routed into a single workspace.

Centralized workspace architecture

In a centralized model, all your security logs, telemetry, and operational data are consolidated into one Log Analytics workspace. This is especially effective for organizations with a centralized SOC or where data residency laws allow for cross-border log consolidation. Centralization provides a unified environment where you can monitor, correlate, and respond to threats across the organization without switching between different data views or workspaces.

Consider this business case as an example. A global manufacturing company with offices in Germany, Malaysia, the UK, and Canada centralizes its security operations in a Munich-based SOC. Rather than managing multiple Sentinel instances across regions, they route all logs to a single Sentinel workspace in the West Europe region. With this centralized architecture, the company simplifies their detection logic, incident triage, and automation workflows.

Design considerations and trade-offs

While centralized architecture offers visibility and operational simplicity, it also brings technical and financial considerations, such as the following:

- **Data ingestion costs**: Cross-region data ingestion incurs egress charges, especially when telemetry is collected from globally distributed endpoints

- **Latency and detection timeliness**: Log delivery from distant regions may experience slight delays, impacting real-time analytics or incident response SLAs
- **Access control complexity**: Centralization brings many teams into a single workspace, requiring granular RBAC and table-level access controls to maintain data boundaries
- **Noise and cost management**: To control costs and reduce alert fatigue, the organization must define diagnostic filters, use basic or archived logs for less critical data, and apply transformation rules to preprocess logs at ingestion

Remember, Microsoft explicitly states that moving a workspace to another resource group or subscription is not supported.

When should you choose a centralized architecture?

Centralized architecture is most effective when the following conditions are met:

- You operate under a unified SOC model covering all regions or business units
- Your organization can legally consolidate telemetry across geographies (always review data residency laws)
- You want to standardize detection and response logic across the board
- You aim to maintain centralized governance over analytics rules, workbooks, and automation

While centralized workspace models are well suited for unified SOC operations, not every organization functions under a single umbrella. Some businesses require autonomy at the regional, departmental, or even customer level. In these cases, a decentralized workspace architecture is often the better fit.

Decentralized workspace architecture

Decentralized architecture distributes security operations by enabling each division or team to maintain its own Microsoft Sentinel workspace. This is particularly beneficial in enterprises with regulatory requirements, autonomous IT teams, or diverse business units.

Rather than consolidating data into a single workspace, each team manages its own environment. This method helps you define custom analytics rules, retention policies, automation playbooks, and alert tuning that align with your specific risks, regulations, and workflows.

Imagine a multinational conglomerate operating in healthcare, finance, and manufacturing. Each vertical faces unique security and compliance challenges. For instance, the healthcare division must comply with HIPAA in the US and GDPR in the EU. The financial arm, meanwhile, faces strict internal auditing, long-term logging, and regulatory scrutiny. The manufacturing side is focused on securing **operational technology** (**OT**) environments.

In this case, attempting to centralize their telemetry would introduce compliance barriers and operational friction. By using separate, decentralized Sentinel workspaces, each unit can maintain control, compliance, and agility by tailoring its detection and response mechanisms to its needs.

Design considerations and trade-offs

Let's consider a scenario where a global legal services firm adopts a decentralized model to offer strict client isolation. Each client receives a dedicated Azure subscription and Sentinel workspace managed via Azure Lighthouse. However, they quickly realize that without GitOps, consistency across tenants is unmanageable. By adopting GitHub-based automation, they streamline content deployment while maintaining version control.

This instance outlines the challenges you can face while using a decentralized workspace. The decentralized model trades global visibility for localized control. While this enables compliance and customization, it also creates silos. Correlating threats across tenants becomes more difficult. Incident duplication may occur if the same IOC is observed in multiple environments, and there's no native *single pane of glass* across workspaces unless you integrate tools such as Azure Lighthouse, Microsoft Defender XDR, or Workspace Manager.

Operational consistency is another challenge. With multiple Sentinel environments in play, deploying consistent detection logic, automation rules, or workbooks requires version control. Some organizations solve this by adopting DevOps practices for content-as-code, syncing policies and analytics across tenants via CI/CD pipelines.

From a resourcing perspective, decentralized workspaces often require more overhead. Each SOC team needs expertise in maintaining their Sentinel environment, and each workspace must be monitored for ingestion health, cost optimization, and rule efficacy.

When should you choose a decentralized model?

Decentralized architecture is most effective when the following conditions are met:

- You must comply with data residency or regulatory restrictions that prohibit data aggregation

- Your organization is federated, with independent business units and distinct security teams
- You're an MSSP managing services across multiple clients, each with unique governance requirements
- Localized incident response and alert tuning are needed

That said, decentralization requires governance discipline, tooling maturity, and a strong internal DevSecOps culture to avoid fragmentation and an inconsistent security posture. Decentralized architecture can only be empowering if it's supported by the right operational strategy. Without automation and oversight, it has the risk of becoming a patchwork of disconnected efforts.

The key is to avoid a *set-and-forget* approach. Decentralization doesn't means decentralizing accountability. Your security operations should remain connected by shared playbooks, shared intelligence, and a clear operational governance model that brings consistency without sacrificing flexibility.

Now, not every organization fits neatly into a centralized or decentralized mold. In reality, many security operations require a balance of central oversight without sacrificing local agility. That's where the hybrid workspace architecture comes in.

Hybrid workspace architecture

A hybrid model allows organizations to operate both centralized and decentralized Sentinel workspaces simultaneously. This architecture is typically composed of a central workspace for high-level threat correlation, governance, and reporting, and multiple regional or business-specific workspaces managed independently by local SOCs or IT teams. This dual-layer design is best suited for enterprises spanning multiple regions, business functions, or regulatory landscapes.

Let's consider a multinational retail and logistics company with operations in 12 countries as an example. Their global SOC is based in the US, responsible for detecting and responding to threats across the enterprise. However, local IT and security teams in each country are responsible for managing compliance, local threat intelligence, and immediate incident response. To meet both global and regional requirements, the company deploys Sentinel in a hybrid model.

A central Sentinel workspace, managed by the global SOC, ingests telemetry from shared services such as Microsoft 365, global DNS, and identity providers. Simultaneously, localized Sentinel workspaces in each region ingest region-specific telemetry such as retail POS logs, OT systems, and country-specific infrastructure. This configuration allows the global team to correlate threats across regions, respond to multi-vector attacks, and implement organization-wide detection rules while enabling local teams to meet data residency regulations and fine-tune their security posture independently.

Design considerations and trade-offs

The hybrid model introduces complexity, but it's a trade-off many organizations accept because of the agility and compliance it affords. Successful hybrid deployments have clear rules about which data goes where, who owns which detections, and how response responsibilities are split between local and central teams.

Typically, hybrid deployments follow one of the two patterns:

- **Vertical delegation**: The central workspace receives specific types of logs (such as identity or cloud infrastructure), while decentralized workspaces handle operational or business-specific data
- **Horizontal federation**: Certain tenants or subsidiaries operate fully independently, but a shared workspace exists for cross-tenant visibility, threat intelligence sharing, or coordinated incident response

In both cases, tooling is essential. Services such as Azure Lighthouse, Workspace Manager, and Microsoft Defender XDR multitenant view allow central teams to oversee multiple workspaces without requiring full administrative access to every environment.

Challenges and strategic planning

Running a hybrid architecture isn't without pitfalls. Without rigorous governance, it's easy for content to drift. Analytic rules, workbooks, or playbooks can differ between workspaces, leading to inconsistent detection coverage. To mitigate this, you can implement a DevOps-based content pipeline, managing Sentinel artifacts as code and deploying them through CI/CD workflows to both central and regional environments.

Another key consideration is data duplication and egress costs. If logs are sent to multiple workspaces, perhaps for correlation or compliance, costs can rise quickly. Organizations should clearly document ingestion rules and only duplicate where absolutely necessary, leveraging basic logs, archive tiers, or API-driven enrichment to reduce redundancy. Typically, pick a central workspace and stick to it.

Let's consider a scenario where a major European telecom company adopts a hybrid model to meet GDPR requirements while maintaining centralized threat visibility. Local SOCs can handle incident response and ingested region-specific telemetry. A central workspace is reserved for global threat correlation and executive reporting. Azure Lighthouse enables tier 3 analysts to view all workspaces with read-only access. Finally, detection logic is stored in GitHub and deployed across tenants using CI/CD, ensuring consistency without compromising local flexibility.

In practice, this hybrid operating model delivers the best of both worlds: it respects GDPR-driven data residency by keeping sensitive telemetry and response actions within each region, while still enabling a single, coherent security posture at enterprise scale. Local SOCs retain authority to triage and remediate in their own tenant and workspace, but global leadership maintains consistent oversight through a central correlation layer and standardized reporting.

When should you choose a hybrid architecture?

Hybrid architecture is most effective for the following conditions:

- You need global visibility and meet regional or business unit-level compliance
- Central and local SOCs must operate in parallel, each with a defined scope
- Your environment includes regulated regions, customer isolation, or M&A complexity
- You're an MSSP managing multiple customers but want shared threat intelligence and operational oversight

Overall, hybrid workspace architecture allows organizations to avoid the limitations of both extremes by offering control where it's needed, and autonomy where it's required. But success doesn't happen by accident. Hybrid only works when it's designed with purpose, operated with consistency, and governed with maturity. Clear roles, automated content deployment, and shared threat intelligence frameworks are critical to ensure the architecture scales without splintering. Done well, hybrid architecture becomes more than a compromise; it becomes a strategic advantage.

Cost, access control, and scalability

Whether you lean toward centralized, decentralized, or hybrid, you must understand the implications each approach comes with to avoid downstream friction. Let's break down how these models compare through three critical lenses: cost, access control, and scalability.

Cost: Visibility versus duplication

From a cost perspective, at least on the surface, centralized workspaces are often the most economical. By consolidating all logs into one workspace, organizations can take advantage of committed ingestion tiers, apply uniform retention policies, and simplify billing oversight. You avoid redundant infrastructure and gain economies of scale on ingestion, retention, and query execution.

However, centralization also increases the risk of cross-region data egress. If you're ingesting logs from multiple geographies into a single workspace, you may encounter unexpected charges, particularly if verbose logs such as firewall or proxy data are involved. There's also a hidden cost in over-ingesting irrelevant or unused telemetry because centralized teams often lean toward collecting everything *just in case*.

By contrast, decentralized workspaces can help optimize costs by scoping ingestion to only what each business unit or region needs. Teams can apply specific filters, tune retention based on the use case, and avoid ingesting *noise* into a global platform. But they lose the benefits of consolidation and run the risk of inconsistent billing models and underutilized commitment tiers.

Hybrid models, while flexible, are complex. Duplication of critical logs between local and central workspaces must be tightly controlled, or costs can escalate quickly. Organizations using hybrid models must be disciplined in defining what data is centralized for global insight and what remains localized for tactical analysis.

Access control: Simplicity versus precision

Managing access across teams and regions is a constant balancing act between usability and security. Centralized architectures are simpler to set up from a permissions perspective. You manage access within a single workspace, apply RBAC at the resource group or table level, and monitor interactions centrally. But the downside is that roles may become too broad. Without proper controls, analysts may see more than they should, which raises concerns about data privacy and overexposure.

Decentralized models offer more precise access control. Each team or SOC unit is assigned to its own Sentinel instance, with scoped RBAC that matches operational boundaries. This structure makes it easier to comply with least-privilege principles but also requires more governance. Each workspace needs its own role structure, identity strategy, and approval workflows.

Hybrid deployments require maturity to manage. They typically demand both a shared access framework for central teams and isolated controls for local teams. Tools such as Azure Lighthouse and Workspace Manager are essential to balance visibility and control, but they must be deployed thoughtfully and governed continuously.

Scalability: One size or fit for purpose?

When it comes to scalability, centralized models scale well up to a point. If your telemetry volume remains predictable and your SOC structure is centralized, a single workspace can ingest hundreds of GBs per day (even TBs, when backed by dedicated clusters). You benefit from consistent rule logic, automated deployments, and easy integration into your SOC toolchain.

But once the organization becomes geographically diverse or functionally federated, scalability hits a wall. Routing everything to one place can become a bottleneck, and different teams may need different views of the data or want to move at different speeds. This is where centralization starts to strain.

Decentralized models, on the other hand, scale laterally. You can bring new workspaces online independently, apply localized logic, and maintain performance regardless of what's happening in other business units. However, this also means scaling your operational governance, support model, and content management processes across every workspace, which is something many organizations underestimate.

Hybrid architecture shines here when done right. You get the ability to scale both globally and locally, supporting high ingestion use cases in one part of the business while maintaining lean deployments in others. But this scalability comes with a cost: the operational overhead of keeping everything aligned. That means CI/CD pipelines, shared repositories, automation rules, and clear ownership models across teams.

Making the right call

Ultimately, there's no single best architecture, only what's best aligned with your business needs. If you value simplicity and centralized control, start with a single workspace and optimize for scale. If autonomy, data sovereignty, or multitenant customer delivery is your priority, consider decentralization. If you need both, be prepared to design and invest in a hybrid approach that grows with you.

Whichever model you choose, treat architecture as a living decision and not a one-time choice. Your Sentinel workspace strategy should evolve alongside your team's capabilities, your data landscape, and the threats you're trying to protect against.

Designing role-based access control (RBAC) for Microsoft Sentinel

RBAC enables secure, efficient collaboration, supports segregation of duties, and enforces the principle of least privilege. It does this by assigning permissions to roles (job functions) rather than individuals and then scoping those roles to only the resources and actions required.

RBAC offers the following benefits:

- **Secure, efficient collaboration**: Teams can access the same platform without sharing accounts or over-provisioning; access is repeatable and auditable
- **Segregation of duties**: Different roles are granted distinct capabilities (e.g., read-only monitoring versus rule changes versus response actions), preventing one person from controlling the entire process
- **Least privilege**: Users get the minimum permissions needed, limited by scope (tenant/subscription/resource group/workspace) and removed or adjusted as responsibilities change

When done poorly, it leads to friction, overexposed data, or, worse, undetected misuse or unauthorized changes that open the door to internal and external threats.

RBAC is not just about who can log in; it also defines who can investigate, who can respond, who can manage rules, and who is ultimately accountable. In other words, RBAC reflects your operating model, and designing it thoughtfully is essential to achieving both agility and assurance.

Let's break down how to architect RBAC for Microsoft Sentinel so that it empowers your teams, without compromising security or oversight.

The role model: How access works in Sentinel

RBAC in Microsoft Sentinel is built on Azure's native access control model. Permissions are applied to Azure resources, and access can be scoped at various levels, from the entire subscription down to individual resource groups, or even tables within Log Analytics.

Sentinel-specific access is governed by roles assigned at the Log Analytics workspace level. The key roles most relevant to Sentinel deployments include the following:

- **Microsoft Sentinel Reader**: View-only access to the Sentinel blade, incidents, and analytics rules
- **Microsoft Sentinel Responder**: Can update incidents such as assigning, closing, and adding comments, but cannot modify rules or configurations
- **Microsoft Sentinel Contributor**: Full operational access to configure analytics rules, playbooks, and workbooks, but not to manage workspace settings
- **Log Analytics Contributor**: Can read and write to the underlying workspace, but may have more access than required for Sentinel-specific duties
- **Reader/Contributor/Owner (Azure native)**: Can be assigned at the subscription or resource group level but should be used with caution in Sentinel contexts

The best practice is to avoid using overly broad roles unless absolutely necessary. Use Sentinel-specific roles wherever possible to align responsibilities cleanly and minimize lateral risk.

Practical application: A tiered SOC model

Let's take a mid-sized enterprise with a growing SOC team. They've implemented Microsoft Sentinel centrally and operate on a follow-the-sun model, with analysts in North America, Europe, and Southeast Asia.

To avoid confusion and overlapping in incident response, they assign roles by SOC tier:

- Tier 1 analysts are granted the Sentinel Responder role. Their job is to triage and escalate, not to configure rules or edit playbooks.
- Tier 2 analysts and incident handlers receive the Sentinel Contributor role. They're trusted to refine analytic rules, tune alert thresholds, and deploy playbooks for automation.
- Their DevSecOps team, responsible for CI/CD pipelines and detection engineering, is granted Log Analytics Contributor access but only in non-production environments.
- Administrative roles, such as Owner or Contributor, at the subscription level are reserved for platform and security leads, strictly controlled under PIM with JIT access enforced.

This model allows the organization to scale securely. No one has more access than necessary, and access rights reflect actual job function, not convenience.

Role assignment pitfalls to avoid

One of the most common mistakes seen in the field is assigning subscription-level access "just to get things working." It may solve short-term permission errors, but it creates long-term risk. For instance, a junior analyst investigating alerts should never have access to modify automation rules or workspace configurations across production.

Another pitfall is not revoking access. Without automated reviews or role expiration, users who no longer need access sometimes retain it long after they've left the organization. To mitigate this, integrate access reviews and PIM into your RBAC governance life cycle.

Now that you've understood how access works in Sentinel and how to choose and assign roles for your organization, let's move on to building an RBAC Lifecycle.

Building an RBAC life cycle: From assignment to expiry

Treat access as dynamic, not static. Your RBAC governance model should evolve just like your environment does. Here's what a healthy life cycle looks like:

- **JIT access with PIM**: Use Microsoft Entra PIM to provide temporary elevation for sensitive roles such as Sentinel Contributor or Automation Operator. Users must request access, justify the need, and automatically time-box their permissions. This reduces standing privilege and provides an audit trail.
- **Automated access reviews**: Implement access reviews available in Microsoft Entra ID Governance for all high-privilege roles and groups. Set reviews to run periodically (e.g., quarterly) and assign resource owners to confirm whether access is still necessary. If not, access is automatically revoked. No manual cleanup is required.
- **Role expiry and alerts**: Where possible, assign access with defined expiration and configure alerts for inactive privileged accounts. Combine this with monitoring tools such as Microsoft Sentinel analytics to flag anomalies in role use or elevation frequency.
- **Audit and reporting**: Continuously track who has access, what level, and for how long. Use workbooks or Sentinel dashboards to visualize role distribution and surface any anomalies or violations of least-privilege principles.

In decentralized or hybrid architectures, particularly in scenarios where you're managing multiple Microsoft Sentinel workspaces across different Azure tenants as in MSSP or multi-subsidiary enterprise environments, traditional RBAC models reach their limits. This is where Azure Lighthouse becomes essential. It enables you to delegate access across tenants without compromising role boundaries, visibility, or control.

Managing access at scale with Azure Lighthouse

Azure Lighthouse allows a managing tenant, such as a central security team or service provider, to operate across multiple customer or business unit tenants using RBAC at the subscription or resource group level, without requiring tenant-to-tenant identity federation. We will discuss a full breakdown of Azure Lighthouse in *Chapter 7*.

This model lets you enforce role boundaries per tenant, so each client or business unit retains control over their own environment. It also centralizes operations, enabling tier 2 or tier 3 SOC analysts to respond to incidents across tenants, without reassigning roles or duplicating accounts. Auditing and reporting can be managed centrally, maintaining visibility across customer workspaces while still respecting data sovereignty and compliance.



Use **Azure Lighthouse + Microsoft Entra PIM** to provide **JIT role elevation** across tenants, minimizing standing access even in complex environments.

When should you use custom roles?

While custom roles bring advantages, they also introduce complexity and potential misconfiguration. Keep it simple: use built-in roles by default, and only create custom roles when you have a proven access gap that built-in roles cannot meet (for example, precise table-level restrictions or tightly segmented SOC duties), then treat the role as code and document it to prevent drift. Only create them when built-in roles lack the necessary granularity. You need table-level access, such as isolating analytics from sensitive data, to create these roles. Use them when you're managing cross-functional SOCs with highly specific duties.



If you use custom roles, **document them rigorously**, and manage them via source control (e.g., Bicep or ARM templates) to avoid drift across tenants.

Segment access by scope

Be intentional about what access is granted at what scope. Remember, scope matters because it is your primary blast-radius control: by granting permissions only at the lowest practical level (management group, subscription, resource group, workspace, or even specific resources), you prevent unnecessary lateral visibility and changes, reduce the impact of compromised accounts or mistakes, and keep regional/compliance boundaries intact while still enabling the right teams to

operate effectively. Use resource group-level delegation for tenant isolation and workspace-level scoping for teams managing specific Sentinel instances. Also consider table-level RBAC if you need fine-grained controls on sensitive log data.

Audit, rotate, review

Pair Lighthouse with access reviews and privileged role expiration via Entra PIM. This would run periodic reviews to verify that analysts still require access across all assigned tenants. The pairing will expire unused permissions and automatically clean up access after project or contract completion. It also logs and monitors Lighthouse activity using Sentinel to detect unusual access behavior across tenants.

Let's consider an example scenario. A managed security provider supporting over 40 customers has adopts a scalable and secure approach by assigning each client a dedicated Microsoft Sentinel workspace. Through Azure Lighthouse, tier 1 SOC analysts are granted Reader or Responder access within individual customer tenants, allowing for localized alert triage and response. Tier 3 analysts, on the other hand, have read-only access across all tenants to enable centralized threat hunting and escalation.

All Sentinel content, such as analytics rules, playbooks, and connectors, is managed as code in GitHub and deployed across environments using CI/CD pipelines, ensuring consistency and version control. To uphold trust and compliance, the provider conducts automated access reviews every 90 days and limits long-term Contributor access through Microsoft Entra PIM, reducing standing privilege across the environment.

Now that you've learned how to efficiently design RBAC for Sentinel, let's move on to the Microsoft Sentinel pre-deployment checklist.

Sentinel pre-deployment checklist: Planning with precision

Microsoft provides a pre-deployment framework for Sentinel to help organizations establish a solid foundation that ensures technical readiness, organizational alignment, and operational clarity. In this section, we will discuss that framework, refined with field-tested insights that separate successful deployments from those that stall or sprawl.

Step 1: Identifying and prioritizing use cases

Sentinel is a versatile platform, but without defined goals, it can become a bottomless data lake. Before you deploy anything, have a clear idea of what problems you're trying to solve. Are you focused on identity-based threats? Want to detect lateral movement? Need visibility into cloud workloads or OT environments?

Many successful deployments start with just a handful of use cases tied to known gaps or risk areas. From there, they expand iteratively using a *land and expand* model, which validates each phase before scaling further.

Start with the "crown jewels" and business processes:

- What would materially harm the business if compromised: privileged access, customer data, billing systems, core network services, production workloads, or OT control systems?
- Convert each into a detection outcome: "detect unauthorized access to X," "detect privilege escalation affecting Y," "detect lateral movement toward Z."

For example, a telecom provider's "crown jewels" might be Entra ID privileged roles and core network management platforms. That leads directly to identity and privileged access use cases.

Ensure you use real evidence, not hypotheticals. Pull from the following:

- Recent incidents and near-misses
- Purple team/pen-test findings
- Risk register items
- Audit/compliance observations
- Existing tool alert gaps (Defender, Entra, firewall, and EDR)

If you cannot action an alert, deprioritize it or treat it as "visibility only" until operations mature.

Step 2: Estimating ingestion and budget

Sentinel is billed primarily based on data ingestion, so estimating your projected volume is crucial. Use Microsoft's pricing calculator to forecast costs based on your chosen data sources and retention needs.



You can find the pricing calculator at `https://azure.microsoft.com/en-us/pricing/details/microsoft-sentinel/`.

While estimating ingestion and budget, consider the following questions:

- What types of logs are you ingesting (for example, security events, audit logs, or firewall logs)?
- How verbose are those logs?
- How long do you need to retain them?
- Are you using analytics logs, basic logs, or archived logs?
- Would a committed data tier save you money compared to PAYG?

Also think beyond cost: ingestion impacts query performance, incident fidelity, and downstream automation. Logging everything may feel safe, but it rarely delivers value without a strategy.

Step 3: Selecting your workspace architecture

The architecture decision of using centralized, decentralized, or hybrid should be made early, not retrofitted after you're live. Your chosen model will influence log routing and ingestion design, permissions and RBAC structure, CI/CD pipelines for content delivery, and multitenant support if managing more than one organization or customer. If you're unsure, start with centralized for simplicity, but be ready to scale to hybrid when data residency or SOC autonomy becomes a factor.

Step 4: Assembling your deployment team

Sentinel is built by a cross-functional team and not by a single person. At a minimum, the team should include a security architect to define detection and response priorities, a platform engineer or Azure admin to manage workspace and log ingestion, an SOC lead or analyst to validate incident management flow, and a compliance or governance lead to ensure policies are followed.

This team doesn't have to be large, but it does need to be collaborative. Many failed deployments stem not from technical errors but from disconnects between teams that assumed someone else was handling a critical step.

Step 5: Planning Defender and Microsoft 365 integration

Some of Sentinel's most valuable data sources, such as Microsoft Defender for Endpoint, Defender for Cloud, and Entra ID logs, are natively integrated and available within Azure and Defender XDR. Prioritize connecting these early.

One of the most overlooked issues during deployment is the relationship between Microsoft Sentinel and Microsoft Defender for Cloud, especially when it comes to Log Analytics workspaces. Sentinel can only read logs from workspaces that have been explicitly connected to it. If a log is sitting in a workspace Sentinel isn't watching, it's effectively invisible to your SIEM, which means no analytics, no alerts, and no visibility.

This becomes a problem when Defender for Cloud creates its own default workspace, which often happens automatically during onboarding. If that workspace is in a different region or was set up without your knowledge, it may not align with the workspace you're using for Sentinel.

So, for example, consider you deploy Microsoft Sentinel in a central workspace in West Europe. Defender for Cloud, when enabled, creates a new workspace in North Europe. Logs from Defender start going to that new, separate workspace. But because that workspace isn't connected to Sentinel, you won't see any of those signals in your investigations or analytics.

This misalignment is subtle but critical. You could be missing out on security events you think you're collecting.

Step 6: Defining your automation strategy

Automation is one of the most powerful capabilities within Microsoft Sentinel, but only if it's done with intent. Sentinel offers several automation layers, each serving a different purpose:

- Playbooks, built on Azure Logic Apps, can automate everything from sending emails to executing remediation actions.
- Automation rules act as orchestrators. They define when and how playbooks trigger based on alert types, incident severity, or other conditions.
- Scheduled analytics rules can run on a regular cadence, automatically generating alerts or enriching datasets, even when no real-time events are firing.

Align automation with **SOC maturity**. Don't over-automate immature environments as it can erode analyst trust. But don't under-automate either. Even basic tasks, such as alert enrichment, can dramatically reduce analyst fatigue.

Before jumping into Logic Apps and connectors, ask yourself: *What repetitive tasks are draining my team? Which alerts never need human eyes? Where are we losing time or missing opportunities for speed?*

With your initial detection use cases defined, the next question is how you will run them at scale without overwhelming analysts. This is where automation becomes a force multiplier by turning repeatable triage steps and routine responses into consistent workflows that reduce noise, speed up investigations, and keep the SOC focused on the incidents that genuinely require human judgment.

Examples of practical automation use cases include the following:

- **False positive suppression**: Use automation rules to auto-close alerts that match known benign behaviors or indicators such as internal vulnerability scans or allowed external logins
- **Threat intelligence enrichment**: Auto-append WHOIS, GeoIP, or VirusTotal data to alerts before an analyst even opens the incident
- **Remediation workflows**: Automatically block malicious IPs using a firewall rule or isolate compromised devices via Defender for Endpoint
- **Ticketing system integration**: Route high-priority Sentinel incidents to ServiceNow, Jira, or Microsoft Teams for collaborative triage



Even if you're not ready to automate the response, start by automating **context enrichment**. It's one of the highest-ROI/lowest-risk ways to reduce analyst fatigue.

If your SOC is still maturing, avoid over-automation early on. Automatically blocking users or deleting email messages can backfire and erode analyst confidence if the logic isn't 100% solid. Focus on assistive automation instead by enriching data, notifying stakeholders, or flagging anomalies.

As your team builds trust in the system and gains operational consistency, you can graduate toward response automation, using change control and approval gates to reduce risk. The good news is that you don't have to start from scratch. Microsoft and the community have already built hundreds of ready-to-deploy playbooks and automation assets.

Here are a few must-know resources:

- **Microsoft Sentinel GitHub repository**: This is the central hub for everything from playbooks to analytics rules and workbooks. Check out the **/Playbooks** folder to find automation examples for phishing response, Defender integrations, and third-party connectors. Access it here: `https://github.com/Azure/Azure-Sentinel`.

- **Azure Logic Apps Sentinel templates**: There are Microsoft Learn tutorials that guide you step-by-step in building real-world incident response workflows. Access them here: `https://learn.microsoft.com/en-us/azure/sentinel/tutorial-respond-threats-playbook`.

Let's now move on to the next and final step.

Step 7: Documenting governance, roles, and maintenance

Deploying Microsoft Sentinel is one thing, but operating it well over time is another. That's why the final step in your pre-deployment checklist is about codifying governance and ownership. You don't need a 40-page playbook to get started, but you do need a living document that answers one critical question: *Who does what, and when?*

Think of this as your Sentinel operational manual, and it should cover the full life cycle of the platform.

To get started, you can answer the following questions:

- Who's responsible for detection engineering, including creating, tuning, and validating analytics rules?
- How often are rules reviewed, retired, or updated to reflect new threat intelligence?
- What's your alert escalation path, especially for critical or noisy alerts?
- Who manages cost monitoring, evaluates data source value, and decides whether new logs are worth ingesting?
- How is access reviewed, rotated, or revoked to maintain a secure RBAC model?

When these aren't documented, gaps quickly emerge, rules go stale, costs creep up, access lingers, and response times lag. Sentinel ends up being reactive instead of proactive.



Start small. Create a shared doc or internal wiki page. Assign clear roles and review cycles. Revisit it monthly as part of your SOC or security operations review.

Skipping the planning phase may get you to deployment faster, but it guarantees technical debt. Sentinel rewards organizations that invest in upfront design—not with flashy dashboards, but with meaningful, actionable outcomes. The pre-deployment phase is about ensuring that the time, effort, and budget you invest in Sentinel translates into actual security outcomes.

Now it is time to move on to your Microsoft Sentinel workspace configuration and overall settings. Let's begin with naming conventions as a key to achieving consistency within your deployments.

Workspace naming conventions

A well-defined naming convention is important to manage and identify resources efficiently, especially in complex environments. Microsoft recommends incorporating the following key elements into your resource names to convey essential information at a glance:

- **Resource type abbreviation**: Use standardized abbreviations, for example, *law* for *Log Analytics workspace*
- **Workload or application identifier**: Indicates the specific application or service the resource supports
- **Environment indicator**: Denotes the deployment environment, such as *prod* for *production* or *dev* for *development*
- **Region code**: Specifies the Azure region, such as *eus* for *East US* or *wus* for *West US*
- **Instance number**: A unique identifier to distinguish between multiple instances

For example, let's take *law-finance-prod-eus-001* as an example for a naming convention. This name represents a Log Analytics workspace for the finance application in the production environment, located in East US, and is the first instance. Adhering to such conventions facilitates resource management, automation scripting, and compliance auditing.

The next consideration is what happens when those well-structured workspaces start ingesting at serious volume. At that point, scalability and cost optimization move from "nice to have" to design requirements, and this is where dedicated Log Analytics clusters enter the conversation.

Dedicated cluster guidance for high-volume ingestion

As your Microsoft Sentinel deployment matures and log volume grows, scalability becomes a necessity. For organizations regularly ingesting over 500 GB of data per day, deploying a dedicated log analytics cluster is a strategic move that brings performance, cost, and operational benefits into sharper focus.

A dedicated cluster essentially creates an isolated and optimized compute environment to handle Sentinel's data ingestion and query processing. Unlike standard shared infrastructure, dedicated clusters ensure that your workload doesn't have to compete with others, which significantly improves the consistency and reliability of your security operations, especially during peak periods or large-scale investigations. One of the advantages of deploying dedicated clusters is performance optimization. By isolating resources, queries run faster and more predictably. This improves operational efficiency for SOC teams that rely on real-time responsiveness, such as in threat hunting or incident investigations.

Dedicated clusters are cost-efficient as they also enable smarter use of Microsoft's commitment tier pricing. When multiple Sentinel workspaces are connected to a single cluster, their collective ingestion volume can be aggregated, helping you qualify for better pricing bands. This consolidated approach is particularly beneficial for large enterprises or MSSPs managing multiple tenants and workspaces.

This model also improves security. Dedicated clusters support advanced data protection features such as **customer-managed keys (CMKs)** and double encryption. These capabilities are especially valuable in highly regulated industries such as finance, healthcare, and government, where data sovereignty and auditability are top priorities.

Operationally, dedicated clusters offer centralized management capabilities. With multiple workspaces under a single cluster, it's easier to oversee performance, apply consistent configurations, monitor data ingestion patterns, and manage access controls. This not only simplifies day-to-day operations but also strengthens governance.

That said, implementing a dedicated cluster requires deliberate planning. First, conduct a detailed analysis of your current and anticipated data ingestion. If your organization is consistently ingesting logs at scale, or expects to do so as more data sources come online, a dedicated cluster can help prevent future bottlenecks. It's also important to align your deployment with the right commitment tier, which allows you to optimize costs based on your expected ingestion volume. This decision impacts both budgeting and long-term scalability, so it should be made with support from finance, procurement, and technical teams.

When it comes to access control, remember that each workspace connected to the cluster still requires appropriate RBAC settings. If your cluster supports multiple departments, business units, or clients, make sure access is scoped carefully to maintain data boundaries and adhere to least-privilege principles.

Finally, regularly monitor performance metrics using Azure Monitor and Sentinel workbooks. Keep an eye on ingestion latency, query duration, and workspace health. Use these insights to optimize your ingestion patterns and fine-tune your analytics rules and playbooks.

Now, with your workspace architecture firmly in place, the next step is deciding what data you actually bring into Microsoft Sentinel. This is where strategy matters most. Just because you *can* ingest a data source, doesn't mean you *should*. Every log adds cost, complexity, and cognitive load for your analysts. The real goal is to feed your SIEM the right data—not all the data.

Data connector planning in Microsoft Sentinel

In Microsoft Sentinel, data connectors determine what visibility you have in your SIEM, where your detections come from, and how well your response engine performs. But here's the catch: an unstructured approach to ingestion will almost always result in ballooning costs, alert fatigue, and performance degradation.

To get the most value from Sentinel, you need to prioritize signal quality over quantity. Strategic data connector planning ensures your ingestion pipeline is efficient, compliant, and scalable so that your SOC sees what matters without being buried in noise.

Start with what matters: Prioritize high-value data sources

Start by integrating Microsoft-native services. These offer the highest signal fidelity, easiest onboarding, and, in some cases, zero-cost ingestion.

The following are some key high-value connectors to prioritize:

- Microsoft Defender XDR
- Microsoft Defender for Endpoint
- Microsoft Defender for Identity
- Microsoft Defender for Office 365
- Microsoft Defender for Cloud Apps
- Microsoft Defender Alerts
- Microsoft Defender Vulnerability Management
- Microsoft Purview Data Loss Prevention
- Microsoft Entra ID Protection
- Microsoft Threat Intelligence
- Microsoft Entra ID
- Microsoft Defender for Office 365
- Microsoft Defender for Cloud



Use the Microsoft Sentinel content hub catalog to explore and deploy prebuilt data connectors, analytic rules, and workbooks in one click: `https://learn.microsoft.com/en-us/azure/sentinel/sentinel-solutions-catalog`.

The next logical step is to expand your telemetry to include external security infrastructure, such as firewalls and NGFWs (Palo Alto Networks, Fortinet, and Check Point), IDS/IPS (Suricata and Snort), endpoint protection (CrowdStrike, SentinelOne, and Sophos), and network security tools (Cisco Umbrella, Zscaler, etc.). Use Syslog or **Common Event Format** (**CEF**) connectors for these integrations. These connectors are supported via the Log Analytics agent or the **Azure Monitor Agent** (**AMA**), with the latter offering granular **data collection rule** (**DCR**) support to selectively ingest logs. We know that it can sometimes be tempting to ingest everything *just in case*. But remember that Sentinel bills by GB. So, a stream of verbose debug logs can sink your budget and swamp your analysts.

With AMA, you can filter at the source using DCRs. These let you target specific event IDs, such as 4625 for failed logons, log levels such as Warning or Error only, and sources such as only security logs, not system. For example, instead of ingesting all Windows event logs, collect only event IDs 4720, 4726, 4732, and 4624 from domain controllers, which are events related to account creation, deletion, group changes, and logons.



It is worth investing time to learn about DCR filtering as it will improve your data quality, help you scale, and even apply a transformation to filter or modify incoming data before it's sent to a destination. Check out the DCRs here: `https://learn.microsoft.com/en-us/azure/azure-monitor/data-collection/data-collection-rule-overview`.

Use the SOC optimization tool

When built-in connectors fall short, such as for bespoke internal systems, SaaS platforms, or third-party clouds, custom ingestion is your best option.

Custom connector methods include the following:

- Azure functions: Use these for high-throughput, cloud-native log processing with custom logic (e.g., parsing complex JSON from APIs)
- Logic apps: These are great for scheduled or triggered ingestion from low-volume systems or SaaS tools (e.g., pulling alerts from your helpdesk system)
- Logstash: This is a flexible open source tool, ideal for on-premises log parsing and enrichment before sending processed log data to Sentinel
- AMA with custom DCRs: This is useful for pulling logs from flat files, legacy systems, or niche platforms

Align with compliance and retention requirements

Your data strategy must align with legal, regulatory, and internal compliance obligations. Sensitive data such as health records (HIPAA), financial transactions (PCI-DSS), or personal data (GDPR) require careful attention.



Check out this link to find helpful information to understand log cost calculations and options in Azure Monitor: `https://learn.microsoft.com/en-us/azure/azure-monitor/logs/manage-cost-storage`.

Use Azure Monitor's per-table retention policies to define how long each log type is retained. For example, you might retain sign-in logs for 180 days but keep security alerts for 2 years for compliance auditing. Also, ensure that data residency requirements are respected by storing logs in regionally appropriate workspaces or clusters.



Lessons from the field

These are some best practices we've learned from experience:

- Build a monthly or quarterly review cadence to analyze ingestion volumes per connector, review cost reports and anomaly trends, correlate connector performance with alert fidelity, adjust DCRs and retention settings or remove unused sources.
- Use Azure Cost Management, Sentinel workbooks, and the Ingestion Cost Summary workbook (available in the Microsoft Sentinel GitHub repository) to support this analysis.

You've prioritized and ingested the right data. Now it's time to put that intelligence to work. In Microsoft Sentinel, automation is where detection meets action, enabling your SOC to respond faster, reduce manual load, and scale operations without burning out your analysts.

Strategic automation in Microsoft Sentinel

Microsoft Sentinel offers a robust set of automation capabilities that help security teams to streamline workflows, standardize response actions, and accelerate time to containment. Automation in Sentinel typically operates across two key layers: centralizing incident flow and playbooks.

Automation rules: Centralizing incident flow

Automation rules are the brain behind Sentinel's incident response flow. These rules allow you to define conditions such as when an incident is created, updated, or tagged, and specify what actions should happen next automatically.

For example, you can configure Sentinel to auto-assign incidents based on severity or source, tag incidents for triage or escalation, trigger specific playbooks based on incident type, and suppress or auto-close false positives based on known behavior. This ensures your SOC follows consistent, policy-driven processes regardless of who's on shift. It also reduces the risk of human error and makes the response more predictable.



You can chain multiple automation rules with priorities and conditions. Use this to differentiate between enrichment-only playbooks versus full-blown remediation flows.

Playbooks: Orchestrating response with Logic Apps

While automation rules define when to *act*, playbooks define *how*. Playbooks are built on Azure Logic Apps, and they let you create multi-step workflows that span across Microsoft and third-party systems. These can be triggered by automation rules or executed manually by analysts.

Common playbook actions include the following features:

- Enriching incidents with WHOIS data, VirusTotal reports, or Microsoft Threat Intelligence lookups
- Notifying analysts or business stakeholders via Teams, Slack, or email
- Creating tickets in systems such as ServiceNow or Jira
- Executing remediation, such as disabling compromised accounts, resetting credentials, or isolating endpoints in Microsoft Defender for Endpoint



Explore 100+ community-driven playbooks in the Azure Sentinel GitHub repository: `https://github.com/Azure/Azure-Sentinel/tree/master/Playbooks`.

Use cases for Sentinel automation

To help ground automation in real-world SOC needs, here are several common and high-value use cases:

- **Incident enrichment**: Automatically pull IP reputation, GeoIP info, and past user behavior into the incident timeline to support faster triage
- **Ticketing integration**: Seamlessly connect Sentinel to ITSM tools such as ServiceNow or Jira so that incident handling follows organizational workflows
- **Communication and escalation**: Trigger alerts to SOC Teams channels, Slack, or email groups when a high-severity incident is created, ensuring rapid visibility
- **Proactive remediation**: For known threats such as ransomware behavior, configure playbooks to auto-disable users, remove access, or trigger Defender isolation on affected hosts



Caution: Start with alert enrichment and notification before moving to auto-remediation. Over-automating early can erode trust if not rigorously tested.

Monitoring the health of your automations

Automation, like any critical system, must be monitored. Sentinel provides execution logs for playbooks and automation rules, allowing you to track success/failure rates, view trigger times and execution duration, and set up alerts on failure, so you know when a remediation flow didn't run as expected.



Use the Automation Health workbook from the Sentinel GitHub repository to visualize the status and impact of all automation over time. For structured automation planning, align your workflows with Microsoft's Azure Security Benchmark (ASB) v3 (`https://learn.microsoft.com/en-us/security/benchmark/azure/`), which is a comprehensive set of best practices based on standards such as NIST 800-53, CIS Controls, and PCI-DSS. ASB provides prescriptive guidance on where and how to automate, including control mappings and example implementations.

Automation in Sentinel isn't about replacing people, but about enabling them to focus on decisions, not distractions. By removing repetitive tasks, enriching alerts in real time, and triggering responses faster than any human can, Sentinel automation acts as a force multiplier for your security team. Start small, build trust, then scale.

As you mature your automation program, the focus naturally shifts from building workflows to running them reliably at scale, across multiple teams, environments, and often multiple tenants. That operational reality is where tenant sprawl becomes the next constraint: you can have well-designed playbooks and strong health monitoring in a single workspace, but sustaining consistent control, visibility, and governance across several Microsoft 365 and Azure tenants requires an explicit multitenant strategy. This is the point where Microsoft's MTO model and the Defender XDR unified SecOps experience become the logical next step.

Multitenant management with Defender XDR and Microsoft MTOs

Managing security across multiple tenants has historically meant juggling disconnected tools, user identities, and alert queues. With the introduction of Microsoft's MTO model and the Defender XDR unified SecOps platform, this complexity is now streamlined into a more cohesive, scalable, and security-focused experience.

Microsoft 365 MTOs are designed for enterprises operating across multiple tenants, whether due to regulatory requirements, acquisitions, or global footprint. They enable organizations to unify user identity, collaboration, and policy enforcement across these environments without needing to collapse them into a single tenant. With MTOs, organizations can synchronize users and manage identities across tenants using cross-tenant synchronization in Entra ID, enable secure cross-tenant collaboration, such as shared Teams channels, calendar availability, and OneDrive access, and apply unified Conditional Access policies and security baselines centrally. Think of MTOs as laying the identity and collaboration foundation, so security operations can be unified without restructuring your entire tenant architecture.

Defender XDR: Centralized security across tenants

Building on this foundation, Defender XDR's multitenant management capabilities allow SOC teams to manage incidents, automation, and investigation workflows across all tenants from a centralized view.

Key capabilities include the following:

- **Unified incident queue**: Defender XDR correlates alerts across identities, endpoints, email, and the cloud into a single incident, even if those signals span different tenants. You can triage, investigate, and act from a single queue rather than pivot between portals.

- **Cross-tenant hunting and investigation**: Using Microsoft Sentinel and Azure Lighthouse, threat hunters can query telemetry across tenants using `workspace()` and `union` in KQL, all from a central workspace. This enables efficient multitenant threat correlation.
- **Centralized automation**: SOC teams can deploy Logic Apps and automation rules from a managing tenant to respond to threats across subsidiaries or customers, so long as appropriate Azure Lighthouse delegation is in place.

Use Microsoft's MTO overview as your reference for architecture, limitations, and supported scenarios: `https://learn.microsoft.com/en-us/unified-secops-platform/mto-overview`.

When paired with Microsoft Sentinel, Defender XDR's multitenant management becomes even more powerful. It creates a centralized Sentinel workspace that can aggregate incidents and telemetry across tenants. Custom workbooks provide executive dashboards with metrics by tenant, region, or business unit. Playbooks can automate responses, push notifications to stakeholders, or sync with ITSM platforms across environments.

Let's consider an example of an MSSP that supports over 50 clients, each with their own tenants. With Defender XDR and MTO, they can use Azure Lighthouse to delegate RBAC-based access to each client's workspace and correlate and triage incidents centrally in Defender XDR's incident queue. It also helps them run cross-tenant queries from a central Sentinel instance to uncover widespread phishing attacks, use Logic Apps to auto-generate alerts in ServiceNow, and notify client contacts via Microsoft Teams. This model provides a unified analyst experience, reduces overhead, and allows the provider to scale securely without duplicating security teams.

Important considerations and limitations

While powerful, multitenant management using MTO and Defender XDR has architectural and operational caveats:

- **Cross-tenant alerts don't propagate automatically**: Cross-workspace rules in Sentinel create incidents only in the central workspace and not in customer tenants. If regulatory requirements mandate local incident visibility, you'll need to configure per-tenant rules or forward summaries.
- **Automation scope is limited by permissions**: Playbooks running in the MSSP tenant cannot perform actions in customer tenants unless the logic app has been explicitly granted permission, typically via Lighthouse delegation.

- **Collaboration complexity**: Teams and SharePoint cross-tenant features require cross-tenant access policies and may not be fully seamless without careful configuration.
- **Query and performance limits**: Running `workspace()` queries across many tenants may introduce latency. Microsoft recommends dedicated clusters for ingest volumes over 500 GB/day.

Follow these best practices for a successful integration of MTO and Defender XDR:

- Designate a managing tenant for your Sentinel and Defender XDR operations
- Use Azure Lighthouse to grant fine-grained access without needing to create accounts in each tenant
- Deploy custom RBAC roles to restrict actions based on analyst responsibility (e.g., tier 1 triage versus tier 3 investigation)
- Use per-tenant analytics rules only where regulatory or business requirements demand localized detection

MTO and Defender XDR together represent a shift for multitenant operations. Whether you're a global enterprise managing regional entities or an MSSP with hundreds of customers, this model gives you the tools to centralize what matters, decentralize where required, and unify your security operations, all without compromising on governance or visibility.

So far, we've explored how Microsoft Sentinel and Defender XDR empower centralized visibility and multitenant threat detection. But visibility alone isn't enough. Standardizing operations at scale is just as crucial. For MSSPs and large organizations managing dozens (or even hundreds) of Sentinel workspaces, keeping configurations aligned, content up to date, and governance intact can quickly become overwhelming. That's where Workspace Manager comes in. It is a new tool designed to bring consistency, automation, and centralized control to Microsoft Sentinel deployments at scale.

Workspace Manager for MSSPs

Workspace Manager is a preview feature specifically designed to streamline large-scale Sentinel operations, especially in environments managed by MSSPs or enterprises with federated teams. It acts as a command center for managing Sentinel workspaces, enabling SOCs to apply updates, push content, and enforce standards across tenants without having to log into each one individually.

With Workspace Manager, MSSPs and centralized SOC teams can perform the following functions:

- Register and manage multiple Sentinel workspaces from a single interface, providing a clear overview of tenant health, ingestion metrics, and configuration drift
- Deploy detection content at scale, including analytics rules, automation rules, workbooks, and playbooks, by using configuration as code, CI/CD pipelines, or GitHub integration
- Ensure version control and standardization by detecting drift and reapplying approved configurations across all workspaces
- Integrate with DevOps workflows using YAML-based templates to streamline updates and automate deployments

Imagine an SOC managing 50+ Sentinel instances across different tenants. Without Workspace Manager, updating a detection rule or deploying a new playbook requires manual effort or script-based pipelines, often leading to version drift and inconsistent protection. With Workspace Manager, the SOC team simply updates the central GitHub repository. Workspace Manager detects the change and automatically synchronizes it across all registered workspaces, ensuring uniform detection coverage in minutes, not days.

Workspace Manager is built to simplify and scale. It supports bulk operations such as onboarding new tenants, pushing analytics rules, and maintaining consistency across environments. It enables centralized governance, reducing the time and risk associated with manual updates. Workspace Manager also includes a monitoring dashboard that gives visibility into ingestion rates, incident volume, and overall workspace health so your SOC can spot and respond to issues proactively.

But remember that Workspace Manager is still in public preview, so its capabilities are evolving. Currently, it requires Azure Lighthouse delegation to operate across tenants, which means that tenants must explicitly grant access. Changes made from Workspace Manager don't retroactively apply unless re-triggered or re-deployed. Also, advanced GitOps features such as automatic content syncing or workspace-specific role-based overrides are in development but not yet fully supported.



Lessons from the field

Use Workspace Manager in combination with a source-controlled GitHub repo or Azure DevOps project. This lets you manage all analytics content, playbooks, and automation logic as code and deploy them reliably using pipelines.

Workspace Manager bridges the gap between visibility and control. For any MSSP or enterprise scaling their Microsoft Sentinel footprint, it offers a way to operationalize best practices, eliminate manual drift, and build trust in their detection content across every tenant they manage.

Next, let's explore the journey of implementing the Microsoft Security suite, with a focus on Microsoft Sentinel.

Deployment roadmaps: Migration and modernization of your SIEM solution

Transitioning to Microsoft Sentinel is not just about replacing an old SIEM. It's a chance to modernize your security operations, simplify tooling, reduce costs, and align with Microsoft's extended XDR capabilities. Whether you're building greenfield, migrating brownfield, or managing hybrid environments, this section provides a complete deployment roadmap.

The right approach to migrate to Microsoft Sentinel depends on your current SIEM posture, cloud maturity, data estate, and business drivers. Microsoft supports three core deployment paths, each offering flexibility and tailored value depending on your organization's context. Microsoft Sentinel deployments typically follow one of three proven paths, each shaped by the organization's starting point and risk appetite: greenfield (building fresh with no legacy SIEM constraints), brownfield (migrating from an established SIEM and preserving what works), and side by side (running Sentinel in parallel to de-risk change and validate outcomes). The sub-sections that follow break down each path in practical terms—who it is best suited to, the benefits and trade-offs, and the implementation steps that will help you move from design decision to operational reality.

Greenfield deployment

A greenfield deployment involves implementing Microsoft Sentinel in a brand-new environment, without dependencies on a legacy SIEM platform. This approach is ideal for organizations adopting SIEM for the first time, greenfield cloud-native workloads, new business units or acquisitions, and security teams looking to build modern operations from the ground up.

Deploying greenfield incurs no technical debt from legacy SIEM configurations. It comes with the benefit of a fully cloud-native architecture with scalable, serverless operations and easy alignment with Microsoft's Zero Trust, XDR, and DevSecOps principles.

Follow these steps for greenfield deployment:

1. Start with a single Log Analytics workspace in your preferred Azure region, following your cost and RBAC boundary structure.
2. Prioritize built-in connectors (e.g., Entra ID, Defender for Endpoint, or Microsoft 365) via the content hub for instant threat coverage.
3. Leverage automation rules and playbooks from day one. Sentinel's SOAR is native; no third-party license is needed.

Next, let's look at brownfield migration.

Brownfield migration

Brownfield refers to migrating from an existing SIEM, such as Splunk, QRadar, ArcSight, Elastic, or Sumo Logic, to Microsoft Sentinel. Brownfield SIEM migration is typically the most complex because you are replacing an operational security capability without breaking detection coverage, compliance reporting, or SOC deploying Sentinel. The complexity usually comes from the following:

- **Telemetry re-engineering**: Legacy SIEMs often have years of bespoke parsing, normalization, and enrichment. Recreating equivalent fidelity in Sentinel (DCRs, AMA, Syslog, CEF, custom logs, and ASIM) takes engineering effort and testing.
- **Detection logic translation**: Correlation rules rarely map 1:1. You must reinterpret logic, thresholds, joins, and time windows into KQL, then tune it against different data schemas and event timing.
- **Operational dependency**: Dashboards, reports, and alert queues are embedded in SOC processes, SLAs, and audit routines. Changing the tool changes how analysts work, how incidents are escalated, and how evidence is produced.
- **Parallel run and validation**: You often need a side-by-side period to prove parity (or improvement) in alert fidelity, which doubles ingestion, monitoring, and tuning effort temporarily.
- **Data retention and compliance constraints**: Regulated environments may require retention, eDiscovery, chain of custody, and historic search expectations that differ between platforms and impact architecture and cost.
- **Integrations and automation**: Ticketing, SOAR playbooks, threat intel feeds, configuration management database (CMDB), and case management integrations must be rebuilt and regression-tested.

- **Stakeholder and change management**: Migrations affect SOC, infrastructure, identity, network, cloud, and governance teams, each with competing priorities and approvals.

The ROI can be high because a successful migration lets you rationalize noisy use cases, reduce duplicated tooling, modernize detections, and standardize operations. But the path is complex because you must preserve service continuity while transforming the underlying system. It's ideal for organizations ready to retire legacy platforms, businesses seeking cost optimization and unified security operations, and enterprises looking to consolidate tools and workflows.

Brownfield migration allows retention of existing detection logic and response processes. It also enables structured migration without starting over, and provides the opportunity to rationalize and modernize detections.

Follow these steps to implement brownfield migration:

1. Use the **Microsoft Sentinel Migration and Modernization Program** (**MSMMP**) to get partner support, tooling, and funding.
2. Map existing detection rules using Microsoft's SIEM parity mapping templates.
3. Use Uncoder.io or Microsoft's rule converters to translate correlation rules into KQL.
4. Prioritize migration of high-fidelity alerts, not everything. Remember, log volume ≠ log value.



Leverage Microsoft's migration guides for Splunk, QRadar, and ArcSight to align legacy rules, automation, and ingestion formats with Sentinel: `https://learn.microsoft.com/en-us/azure/sentinel/migration`.

Side-by-side strategy

The most widely recommended approach for larger or regulated environments is the side-by-side migration model. This allows Sentinel to run in parallel with the legacy SIEM, enabling gradual rollout, validation, and transition of workloads.

This strategy is ideal for complex SOCs with mission-critical detections, organizations with strict compliance or service uptime SLAs, and environments with a mix of on-premises and cloud workloads. Implementing the side-by-side strategy does not cause any disruption to current SIEM alerts or reporting. It also allows SOC teams to upskill and validate new detections and supports the incremental testing and tuning of Sentinel use cases.

Follow these steps to implement the side-by-side strategy:

1. Run log forwarders (e.g., Logstash, Event Hub, or Syslog) to send telemetry to both SIEMs.
2. Use Sentinel's cross-workspace analytics and incident mirroring to correlate detections.
3. Deploy content hub solutions and gradually replace legacy dashboards with Sentinel workbooks.
4. Validate alert fidelity using rule simulators and MITRE ATT&CK mapping.



Treat side by side as a learning phase. Capture feedback from analysts, tune rules weekly, and only decommission legacy components when business and detection coverage are aligned.

Strategic considerations for all migration paths

No matter what path you take, address these core areas early:

Area	Strategic Action
Architecture	Choose a centralized versus decentralized workspace model aligned to RBAC and cost strategy
RBAC	Define least-privilege roles (Sentinel Reader, Contributor, and Playbook Operator) by function
Ingestion control	Use DCRs to filter noise and apply the right log tier (Analytics, Basic, or Auxiliary)
Automation	Use Logic Apps playbooks from GitHub, content hub, or your DevSecOps pipeline
Cost optimization	Start with free ingestion sources (Defender alerts or Microsoft 365 audit logs) and set ingestion caps during onboarding
Skills transition	Provide KQL, MITRE, and Sentinel training to analysts before full cutover

Table 4.2: Migration path options

Phase-based migration framework

Microsoft recommends a four-phase approach to successfully plan, deploy, and scale Sentinel.

Phase 1: Discovery

Take inventory of all current use cases, data sources, and detection rules. Identify mission-critical assets, compliance needs, security objectives, and map integrations and automation dependencies.

Ask the following questions right after discovery/baselining, once you have a clear picture of your current SIEM's data sources, rule catalog, alert volumes/fidelity, compliance needs, integrations, and SOC operating model. This is the point where you can replace assumptions with evidence and avoid migrating noise:

- Which SIEM use cases must be preserved?
- What compliance standards (e.g., ISO or PCI DSS) apply?
- What are the ingestion volumes per log type?
- What is the role and responsibility distribution across the SOC?
- Are SOAR workflows dependent on external systems or scripts?

You can then turn the answers into a prioritized migration backlog (must/should/could), confirm the required telemetry, design RBAC and RACI, and run a side-by-side pilot to validate alert fidelity before scaling.

Phase 2: Design

In this phase, you have to choose your workspace structure (single versus multitenant). Apply consistent naming, RBAC, and tagging policies, and plan log tiering (Analytics versus Basic versus Auxiliary). Use Azure Lighthouse for delegated multitenant management and build an integration strategy using the content hub and DevOps CI/CD pipelines.

Phase 3: Implementation

In the third phase, you will enable connectors in stages, starting with native Microsoft services. You'll also have to apply DCR filtering and daily ingestion caps to control cost. Deploy analytic rules and playbooks with GitHub/Azure DevOps integration, and use templates and smart deployments via Workspace Manager (preview). Also, validate detection rules using test scenarios and historical log backfill.

Phase 4: Operationalization

In the last phase to deploy Sentinel, tune detection rules based on SOC feedback. Implement automation rules to reduce alert fatigue and monitor ingestion, retention, and cost via Sentinel workbooks. You must track coverage against MITRE ATT&CK using built-in dashboards, as well as scheduling quarterly reviews for detection efficacy and automation health.

There is genuine complexity in SIEM migrations, so you should leverage as much support as possible. MSMMP is specifically intended to provide structured partner support (and, where eligible, Microsoft-backed investment) to accelerate planning and delivery.

Microsoft Sentinel Migration and Modernization Program (MSMMP)

MSMMP is a Microsoft-led initiative that offers partner-led assessments and migration support, SIEM rule parity mapping across platforms such as Splunk, QRadar, and ArcSight, technical toolkits for rule conversion, ingestion setup, and historical data streaming, as well as power over ethernet (POE) validation for enterprise-scale deployments.



This program was already covered in *Chapter 3*. However, if you want to learn more about this program, check out this link: `https://partner.microsoft.com/en-mt/partnership/partner-incentives/microsoft-sentinel-migration-and-modernization`.

Best practices for migrating from legacy SIEMs

Migrating to Microsoft Sentinel from a traditional SIEM such as Splunk, QRadar, or ArcSight requires more than just technical execution. The process demands a strategic mindset. We'll now discuss a few proven best practices to ensure your migration not only succeeds but also unlocks the full potential of Sentinel's native capabilities.

Start by focusing on detection rule mapping. Microsoft offers SIEM parity guides and conversion tools that help translate existing rule logic into Sentinel's native **Kusto Query Language (KQL)**. Rather than attempting a one-to-one transfer of every legacy rule, prioritize high-fidelity alerts that align closely with your organization's real business risks. This process is an opportunity to rationalize use cases, reduce noise, and modernize your detection logic. A recommended approach is to build a detection inventory sheet that captures rule names, trigger types, associated data sources, and their current operational impact.

If your existing SIEM stores historical data required for compliance, hunting, or backtesting, plan to import historical logs into Sentinel. This can be achieved using tools such as Logstash, Azure Data Factory, or Event Hubs. Where possible, route high-signal logs such as identity authentications, endpoint detections, and firewall data into standard analytics tables, while moving verbose or low-touch data into Auxiliary or Basic log tiers for cost efficiency. Always preserve the original timestamp and context to ensure historical analysis is accurate.

When it comes to cost optimization, Microsoft Sentinel offers multiple levers. Apply DCRs to filter noisy logs before ingestion. Map logs to the correct tier. Use Analytics logs for real-time alerts, Basic logs for rare investigations, and Auxiliary logs for verbose data needed only for auditing

or hunting. Additionally, leverage pre-purchase plans or commitment tiers to control budget fluctuations. Remember that Sentinel is scalable as well as tunable.

For risk-averse migrations, many organizations choose to run their legacy SIEM and Sentinel in parallel during the transition. This allows teams to validate rule accuracy and alert fidelity side by side. Use Sentinel playbooks or Event Hubs to mirror incidents from your legacy SIEM into Sentinel for SOC review. Weekly rule tuning and side-by-side incident analysis can surface detection gaps or redundant alerts early, before decommissioning the old platform.

Lastly, take the opportunity to modernize your detection and response architecture. Microsoft Sentinel supports native UEBA, integrated Security Copilot AI assistance, and a continuously growing content hub of out-of-the-box detection rules, hunting queries, dashboards, and automation workflows. Instead of porting legacy processes over unchanged, ask yourself: *Can we do this better in Sentinel?*

Essential tools and resources

To support your migration journey with Microsoft Sentinel, Microsoft and the security community offer a robust ecosystem of tools, frameworks, and knowledge bases. These resources are designed to accelerate planning, automate rule conversion, enhance detection capabilities, and ensure operational alignment across every phase of your SIEM transition. Whether you're building new detection logic, migrating existing use cases, or optimizing for cost and coverage, the tools given in *Table 4.3* are invaluable accelerators:

Tool	**Purpose**	**Link**
Migration workbook	Track migration tasks and gaps	`https://learn.microsoft.com/en-us/azure/sentinel/migration-track`
SIEM rule parity sheets	Compare detection rules across SIEM platforms	`https://learn.microsoft.com/en-us/azure/sentinel/siem-migration`
Detection rule converters	Translate SPL (Splunk), ArcSight, and QRadar rules to Sentinel KQL	`https://learn.microsoft.com/en-us/azure/sentinel/migration-splunk-detection-rules`
SOC prime threat detection marketplace	Community-curated library of detection rules for Sentinel, mapped to MITRE	`https://socprime.com/rs/search-result`

Tool	Purpose	Link
KQL search	Discover community-built Kusto queries and hunting use cases	https://www.kqlsearch.com/
Microsoft Sentinel GitHub	Download playbooks, detection rules, parsers, workbooks, and solutions	https://github.com/Azure/Azure-Sentinel
Uncoder.io	Convert detection logic from other SIEMs into Sentinel-friendly KQL	https://uncoder.io/
Content hub (in Sentinel)	Discover Microsoft-verified solutions, rules, and visualizations by vendor	In Microsoft Sentinel | Content Hub
SOC process framework	Align SOC operations (SLAs, roles, and processes) with Sentinel capabilities	https://techcommunity.microsoft.com/t5/microsoft-sentinel-blog/what-s-new-soc-process-framework-is-now-live-in-content-hub/ba-p/3590349
MITRE ATT&CK dashboard	Visualize threat detection coverage mapped to MITRE TTPs	https://learn.microsoft.com/en-us/azure/sentinel/mitre-coverage

Table 4.3: Sentinel tools and resources

Data retention and long-term storage

Data retention is one of the most consequential design decisions in a Microsoft Sentinel deployment because it directly shapes both your operational capability and your long-term cost profile. Sentinel does not just "store logs"; it determines how far back your SOC can investigate, how confidently you can evidence security and compliance requirements, and whether you can spot slow-burning threats that unfold over weeks or months. If retention is treated as an afterthought, teams often default to ingesting everything into Analytics-tier storage, only to discover later that the budget, performance, or governance model does not align with what the business actually needs.

In deployment terms, retention should be defined alongside your use cases, workspace design, and RBAC boundaries because it influences what data you keep, where you keep it, and how quickly it can be queried.

Data ingested into Sentinel is stored in **Log Analytics workspaces**, which come with their own retention cost structure:

- The **first 90 days of retention** are free for Analytics logs
- **Custom table-level retention** lets you fine-tune storage costs
- **Archived logs** provide low-cost long-term storage for infrequent access
- **Basic and Auxiliary logs** are **not eligible** for the 90-day free retention



If you'd like to learn more about Azure Monitor logs cost calculations and options, check out this link: `https://learn.microsoft.com/en-us/azure/azure-monitor/logs/manage-cost-storage`.

Some data is free to ingest, including alerts from Microsoft Defender XDR, Microsoft Defender for Cloud, Azure activity logs, and Office 365 audit logs. Additionally, Microsoft 365 E5, A5, F5, and G5 customers receive 5 MB per user per day for free, and Defender for Servers Plan 2 provides 500 MB per server per day of free ingestion.

You can navigate to **Cost Management | Cost Analysis** in Azure to visualize Sentinel costs by workspace, resource, and service, as shown in *Figure 4.2*:

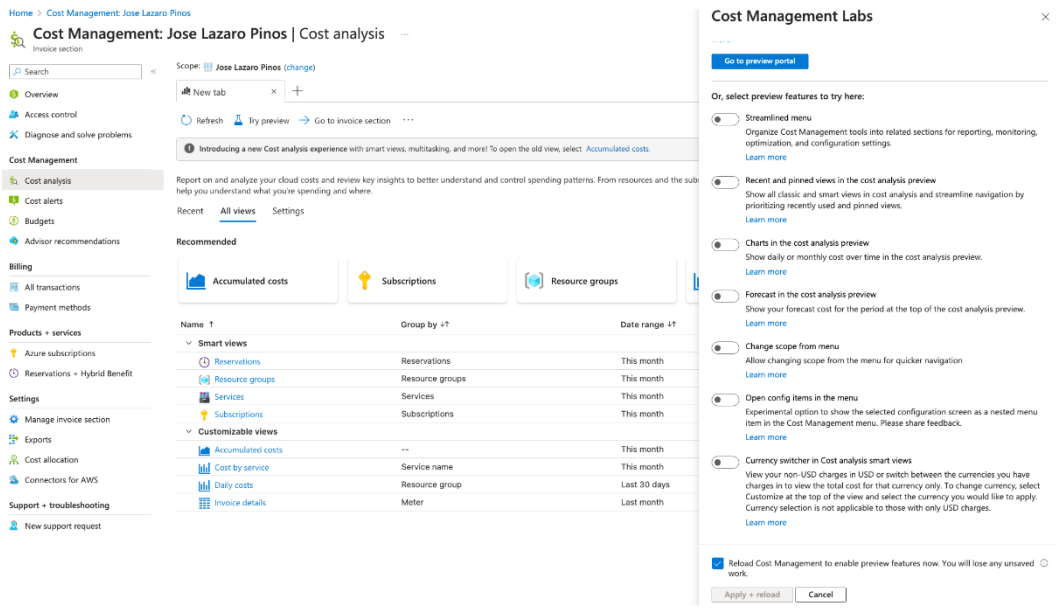


Figure 4.2: Azure – Cost analysis

Optimizing cost, coverage, and performance in Microsoft Sentinel

A use-case-driven ingestion strategy helps optimize cost, coverage, and performance in Microsoft Sentinel. Rather than enabling every connector, start with a detection use case matrix. Map log types to detect scenarios and prioritize logs that trigger high-fidelity alerts or compliance monitoring.

Figure 4.3 shows a detection use-case heat map example:

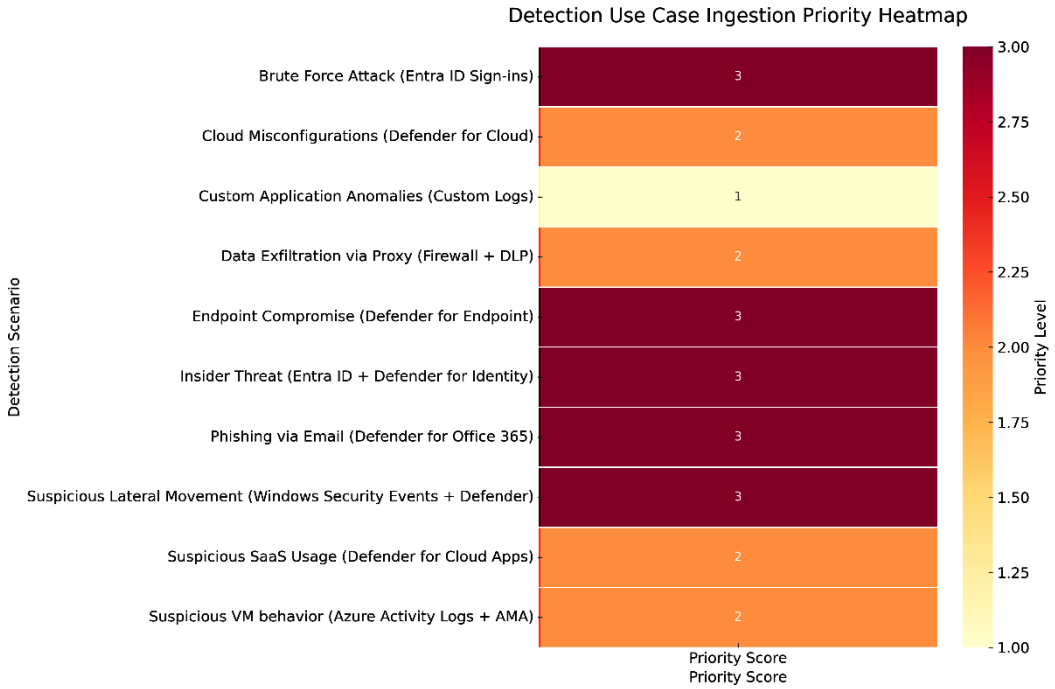


Figure 4.3: Detection use case heat map

Filtering, controls, and budget protection

Filtering, controls, and budget protection are about preventing Sentinel from becoming an uncontrolled ingestion sink. The core principle is simple: decide what you need for detection and investigation, collect only that, and put guardrails in place so costs cannot spike without warning.

With the AMA, DCRs allow you to filter telemetry before it ever reaches Log Analytics. That means you can collect only the Windows Security events that matter for your use cases such as event ID 4625 for failed sign-ins while excluding noisy or low-value events and verbose application logs

that inflate volume without improving detection. For Windows event logs specifically, XPath filtering gives fine-grained control so you can target the exact channels, providers, and event IDs you want, rather than ingesting broad categories "just in case."

Cost guardrails then ensure ingestion stays predictable. Azure Cost Management budgets and alerts give you early warning when spend is trending beyond expectations. Ingestion anomaly playbooks can detect and notify on sudden volume spikes so you can respond quickly (for example, a misconfigured diagnostic setting or a runaway application log). Finally, Azure Policy can enforce standards preventing unapproved diagnostic settings or blocking new high-volume sources from sending data to the workspace unless they meet your rules—so cost control becomes part of governance, not an after-the-fact clean-up exercise.

Table 4.4 compares free and paid log strategies:

Data Placement	**Primary Use Case**	**Cost Model**	**Retention**
Microsoft-provided logs (free)	Audit and compliance-driven visibility	Free	90 days
Analytics tier (Log Analytics)	Real-time threat detection, analytics rules, automation	Higher cost per GB ingested (capacity commitment available)	Extendable
Sentinel Data Lake	High volume of telemetry, historical analysis, AI enrichment	Low-cost storage with separate query and compute costs	Long-term and configurable
Archive or external storage	Compliance, legal hold, rare access to forensic data	Lowest storage cost	Long term

Table 4.4: Log types and cost



Use **Basic or Auxiliary logs** for firewall data, proxy logs, and telemetry that isn't used for alerts but may be needed for investigation or compliance.

The following are tools to support an ingestion strategy:

- **Microsoft Sentinel content hub**: Before enabling any connector, check its associated content pack for analytics rules, workbooks, playbooks, and estimated ingestion volume
- **Sentinel Optimization Workbook**: Use this community-driven tool to optimize costs, threat coverage, and data quality: `https://techcommunity.microsoft.com/t5/microsoft-sentinel-blog/introducing-microsoft-sentinel-optimization-workbook/ba-p/3901489`
- **Cost estimation workbooks**: Preview the impact of a new connector before enabling ingestion
- **Free trials and credits**: Includes 10 GB/day for the first 31 days, Microsoft 365 licensing benefits, and Defender plan ingestion grants

More than just saving money, cost optimization in Microsoft Sentinel is about aligning spending with actual security outcomes. Focus on value-driven ingestion, use automated controls, and leverage free and discounted sources where possible.

Financial journey: Assessing costs from groundwork to full operation

Implementing Microsoft Sentinel and the wider Microsoft Security suite is a strategic investment, one that requires clear-eyed planning to balance performance, coverage, and cost. This section breaks down the financial life cycle of a Sentinel deployment, from early budgeting and licensing to log ingestion, retention, and long-term optimization.

Microsoft Sentinel's pricing model

Microsoft Sentinel is priced primarily on a consumption-based model, meaning your costs are driven by how much data you ingest, how long you retain it, and what additional services you use, such as automation or custom ML models.

What you pay for in Microsoft Sentinel

Microsoft Sentinel costs vary based on several factors:

- **Types of data ingested**, such as Analytics tier versus lower-cost data storage
- **Volume of data ingestion**
- **Retention period** for each data type
- **Additional services**, including Logic Apps, Azure Functions, automation rules, and integrations

With the latest Sentinel consumption model, cost optimization is driven by **intentional data placement**, rather than treating all logs equally.

Sentinel data tiers and usage model

Microsoft Sentinel now uses **three primary data tiers**, each designed for a specific purpose within the detection, investigation, and retention life cycle.

Analytics tier: High fidelity, high value

The Analytics tier is the **premium execution layer** in Sentinel. It is optimized for the following:

- Real-time threat detection
- Analytics rules and scheduled queries
- SOC dashboards and visualizations
- Frequent hunting and automation

These logs provide the most flexible query capabilities and are used directly by detections and response workflows. As a result, they carry the highest ingestion cost and should be reserved for **high signal security data**.

Recommended Analytics tier data includes the following:

- Entra ID sign-in and Conditional Access logs
- Microsoft Defender for Endpoint alerts
- Office 365 audit logs, including Exchange, SharePoint, and Teams
- Azure activity logs

Only ingest data into the Analytics tier when it directly supports detections, automation, or SOC operations.

Sentinel data lake: Cost-optimized secure data storage

With the introduction of the Sentinel data lake, many use cases previously associated with Basic or Auxiliary logs are now better served outside the Analytics tier. The data lake is designed for high volume and verbose telemetry, long-term retention, historical investigations and threat hunting, AI-driven analysis and enrichment, and cross-tenant and cross-dataset correlation. This tier separates storage from compute, allowing organizations to retain large datasets at a significantly lower cost and query them only when required.

Typical data sources include the following:

- Firewall and network security device logs
- VPN, proxy, and secure web gateway telemetry
- DNS and remote access logs
- Infrastructure and application access logs

Archive and long-term retention

For compliance, legal, or forensic scenarios where data is rarely accessed, Sentinel supports archive and external storage options. These provide the **lowest storage cost** and are suited to regulatory retention requirements, legal hold and audit scenarios, and historical forensic investigations.

Important shift from older models

Earlier Sentinel designs relied heavily on Basic and Auxiliary log tiers to reduce ingestion costs. With the data lake, the following apply:

- Basic logs are now less central to cost optimization
- Auxiliary logs are increasingly niche and often unnecessary
- The data lake becomes the default destination for non-detection-driven telemetry



Best practice: Store firewall logs, proxy logs, network telemetry, and other high-volume data in the Sentinel data lake. Promote only high-value subsets into the Analytics tier when they directly support detections, analytics rules, or automation. This approach reduces ingestion costs, improves SOC performance, aligns with unified security operations and Security Copilot, and scales effectively for MSSP and multitenant environments.

Let us look at pricing models and optimization:

- **PAYG**: You're billed per GB ingested, making this ideal for environments with variable data volume.
- **Commitment tiers**: You commit to a daily ingestion threshold and receive discounted per-GB pricing. This offers predictable budgeting but requires accurate forecasting.
- **Pre-purchase plan**: For high-volume deployments, the Pre-Purchase Plan allows you to commit to a set ingestion volume over time, offering the best rates and stability for finance teams.



To learn more about the Sentinel pre-purchase plan, check out this link: `https://learn.microsoft.com/en-us/azure/sentinel/billing-pre-purchase-plan`.

As your Sentinel deployment matures, you may incorporate other Azure services that drive additional costs. *Table 4.5* shows some of the additional services that may incur charges:

Service	**Purpose**	**Pricing Notes**
Logic Apps	Automation playbooks	Charged per action run
Azure Functions	Lightweight data transformation or ingestion	Billed by execution time and memory used
Azure notebooks	Threat hunting and investigation	Compute-based billing
BYOML models	Custom ML-based detection	Requires Azure ML billing

Table 4.5: Additional Sentinel costs

Summary

In this chapter, we covered the critical groundwork required to launch Microsoft Sentinel. From setting up your Microsoft 365 and Azure environments to choosing the right workspace architecture, enforcing access controls, and managing costs, you learned how these foundational steps are essential for long-term operational success.

We also explored how to structure multitenant operations using Microsoft 365 MTO and Defender XDR, giving you the ability to unify security management across customers, subsidiaries, or business units without compromising compliance or scalability. Whether you're deploying Sentinel as part of a greenfield rollout or modernizing an existing SIEM, understanding the financial and architectural considerations upfront is key to making strategic, high-impact decisions. With the core infrastructure in place, the next step is ensuring your Microsoft 365 tenant is fully assessed and hardened before Sentinel goes live.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

Note: *Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



5

Getting Ready for Action: Assess Your Microsoft 365 Tenant

Before deploying Microsoft Sentinel, it's essential to understand your Microsoft 365 environment inside and out. This chapter sets the stage by helping you establish a secure, resilient, and compliant foundation through tenant assessment.

You'll explore how to run three powerful tools built to reveal hidden risks, strengthen your security posture, and support a proactive defense approach.

This chapter covers the following topics:

- Getting ready for action: Assessing your Microsoft tenant with ScubaGear
- Automating tenant monitoring with Maester and Azure DevOps
- Zero trust assessment tool guide

Let's begin by getting your tenant security up to speed with ScubaGear.

Technical requirements

Ensure that you meet the following technical requirements to carry out the processes discussed in this chapter:

- Local administrator rights on your device.
- PowerShell V5 and V7
- Relevant rights across your development/test Microsoft 365 tenant.

Please note that these requirements are kept at a high level as the products continue to change. Please check the updates regarding these regularly.

Getting ready for action: Assessing your Microsoft tenant with ScubaGear

Let's begin by discussing how ScubaGear, developed by the **Cybersecurity and Infrastructure Security Agency** (**CISA**), functions. This tool focuses on ensuring the security of your Microsoft 365 setup aligns with the **Secure Cloud Business Applications** (**SCuBA**) security configuration baseline, a set of security guidelines.

Imagine having a checklist of the best security practices for keeping your Microsoft 365 tenant secure. ScubaGear essentially automates this process by comparing your setup against the SCuBA baseline to ensure everything meets the required standards. This way, you're less likely to miss a critical security setting.

License requirements

To effectively use ScubaGear, your Microsoft 365 tenant must be subscribed to specific security-enabled license bundles, such as E3 or G3, and E5 or G5. These bundles feature key security components such as Microsoft Defender for Office 365 Plan 1 and Microsoft Entra ID P2. Without these licenses, ScubaGear (and other tools) may deem your setup non-compliant with certain security policies.



If you need more info on licensing, please refer to Microsoft 365 Maps (`https://m365maps.com/`), which is the go-to location for license feature information.

Getting ScubaGear up and running

Follow these steps to download, install, and run ScubaGear against your tenant, as well as get the right permissions to run the script.

Step 1: Downloading the ScubaGear script and preparing your computer

Follow these steps to download the ScubaGear script and prepare your system:

1. Visit the official ScubaGear page on GitHub: `https://github.com/cisagov/ScubaGear`.

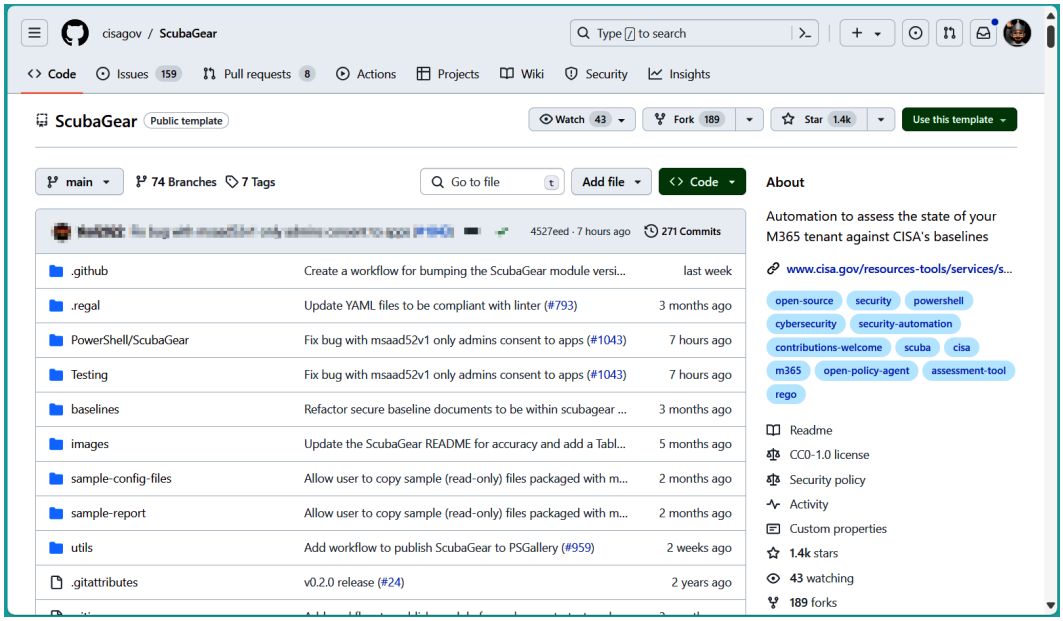


Figure 5.1: ScubaGear landing page

2. Navigate to the **README** section and read through the instructions first. Then, scroll to the **Table of Contents** section and click **Download the Latest Release.**

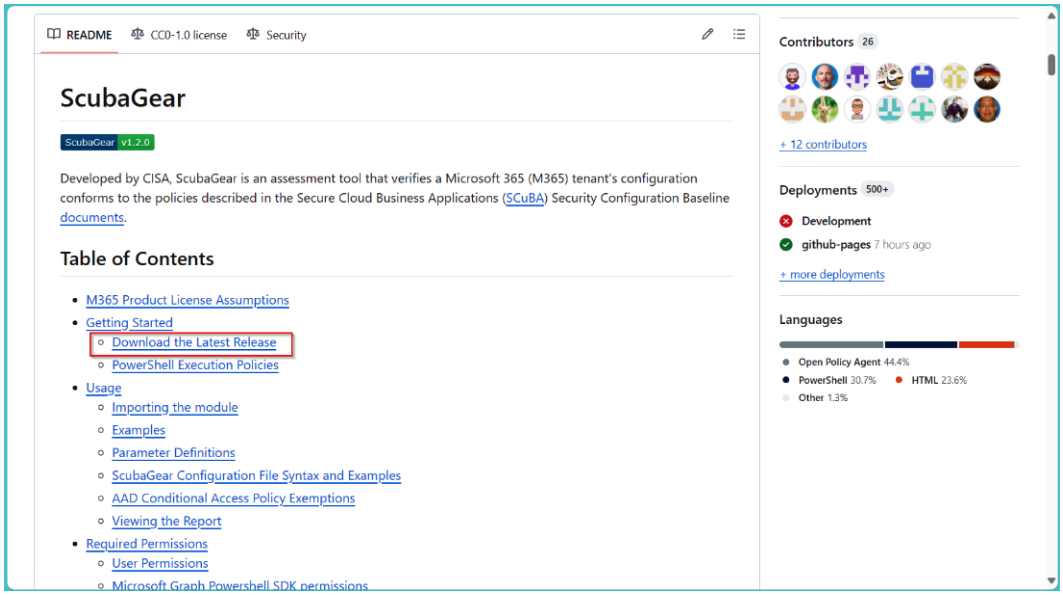


Figure 5.2: ScubaGear Table of Contents page

3. Click the embedded link to see the latest version.

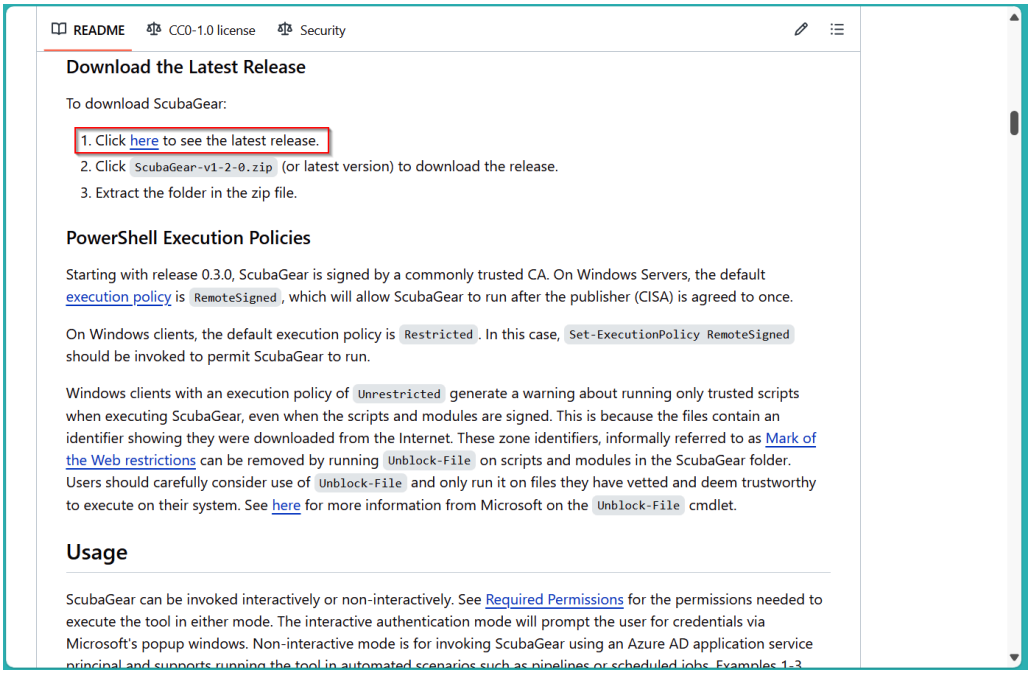


Figure 5.3: ScubaGear download page

4. Download the `.zip` folder of the latest version.



Figure 5.4: ScubaGear Assets

5. Right-click the downloaded `.zip` file and select **Extract**.

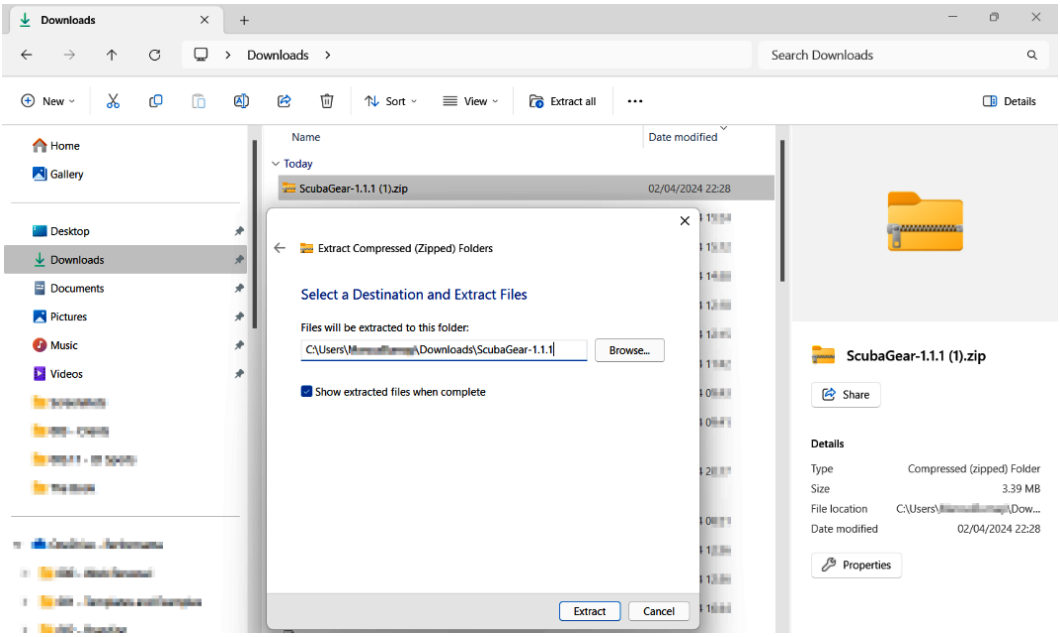


Figure 5.5: Download and extract location for ScubaGear script

Remember the location where you extract the files, such as `C:\Users\UserName\Downloads`.

Step 2: Setting up PowerShell to run ScubaGear

Now, let's ensure that the code can run as it's not officially signed by a publisher. Adjust your execution policy as follows:

- **For Windows servers**: The default **RemoteSigned** setting typically suffices.
- **For Windows clients (PCs)**: Change the default **Restricted** setting to **RemoteSigned**. You can do this by opening PowerShell as an administrator, typing `Set-ExecutionPolicy RemoteSigned`, and pressing *Enter*. Confirm by selecting **Yes to All**.

It's time to set up the PowerShell for your ScubaGear:

1. To make the necessary PowerShell execution changes in PowerShell, you need to run it as an administrator, as shown in *Figure 5.6*.

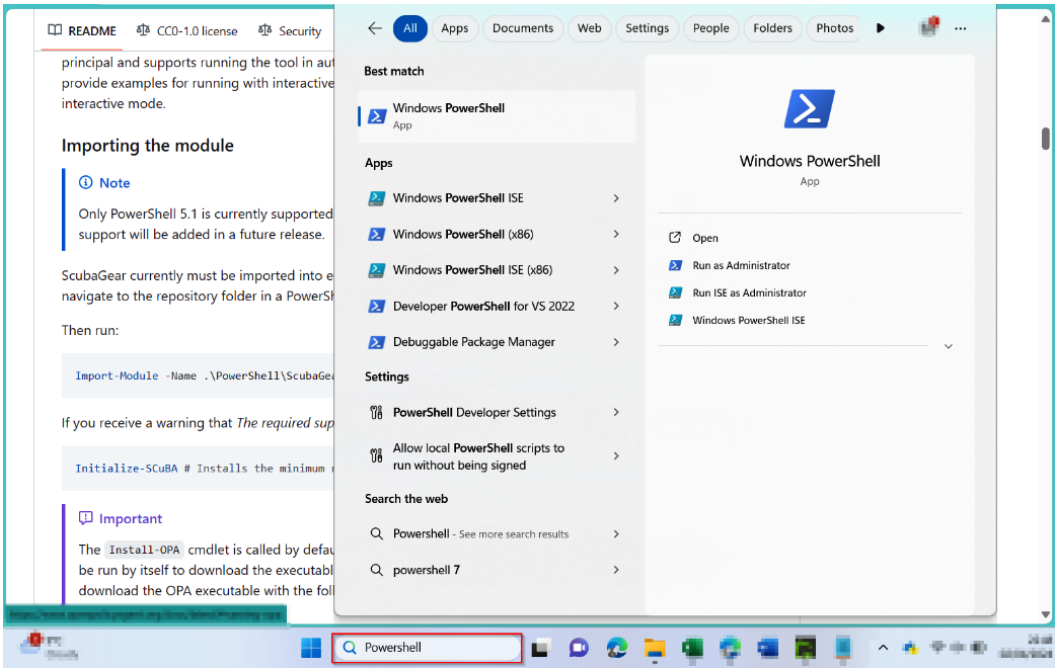


Figure 5.6: PowerShell being executed as an administrator

2. Once PowerShell is open, type in `Set-ExecutionPolicy RemoteSigned` and press *Enter*. This will amend the policy.
3. To set the execution policy for all future requests within this PowerShell session, press *A* to select **Yes to All**, then press *Enter*. Both the instructions and their execution are displayed in *Figure 5.7*.

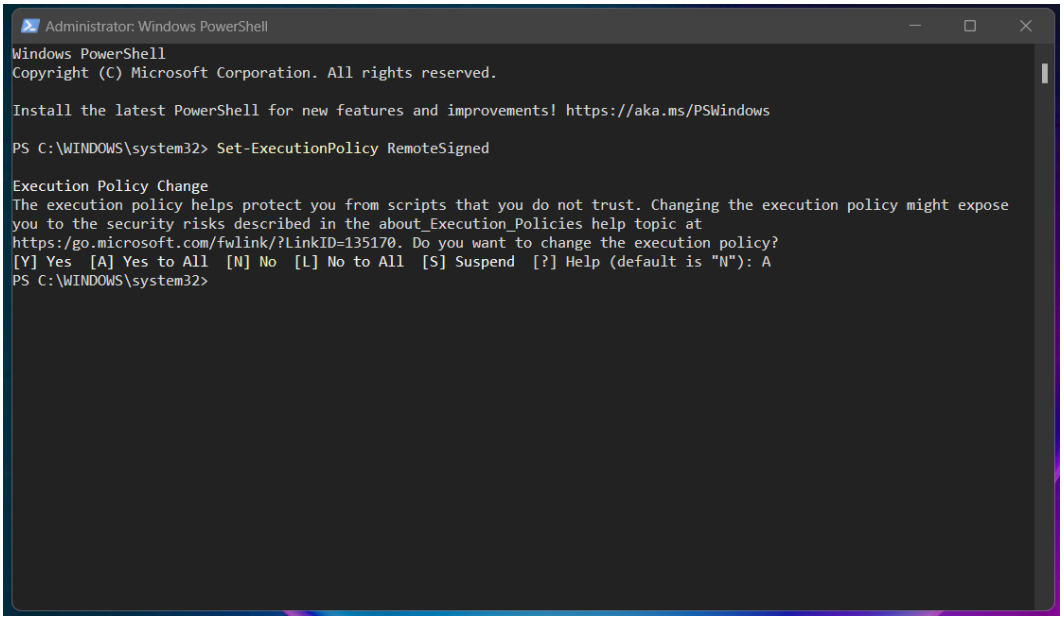


Figure 5.7: Changing the execution policy in PowerShell



Security note: An `Unrestricted` policy may warn you to run trusted scripts only. ScubaGear is signed by a trusted certificate, but caution is advised with any script.

Good to know: ScubaGear can be invoked interactively or non-interactively. We will run this example interactively. The interactive authentication mode will prompt the user for credentials via Microsoft's pop-up windows.

Step 3: Running ScubaGear

Now that you are set up on your local system, let's execute the assessment:

1. Navigate to **ScubaGear Directory | Open PowerShell** and use the `cd` command to navigate to the extracted `ScubaGear` folder. Here is an example:

```
cd C:\Users\UserName\Downloads\ScubaGear-1.1.1\ScubaGear-1.1.1
```

2. Execute the script by running the following command:

```
.\setup.ps1
```

You will see the following after running the script:

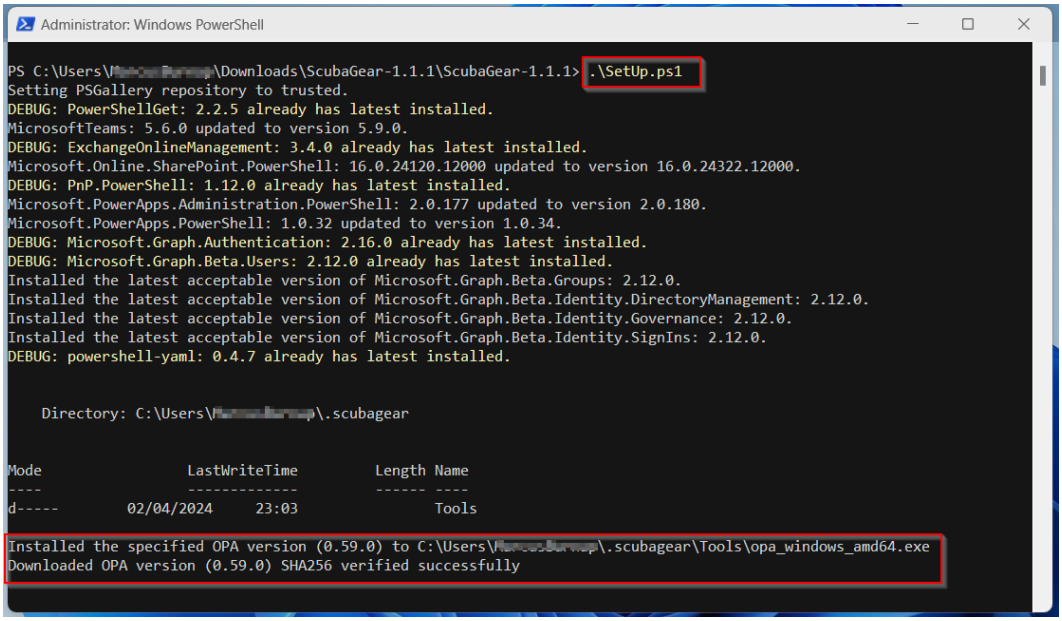


Figure 5.8: Execution of the ScubaGear script in PowerShell

This step installs any required modules. You might see messages indicating the installation status of these modules. A verification message will be presented below, and a new directory location will be created.

3. In this instance, we will run an assessment against all supported products except for Power Platform. Always remember to import the module as the first step, which needs to be completed each time you use this script:

```
Import-Module -Name .\PowerShell\ScubaGear
```

4. Next, run the starting command to call the login popup:

```
Invoke-SCuBA
```

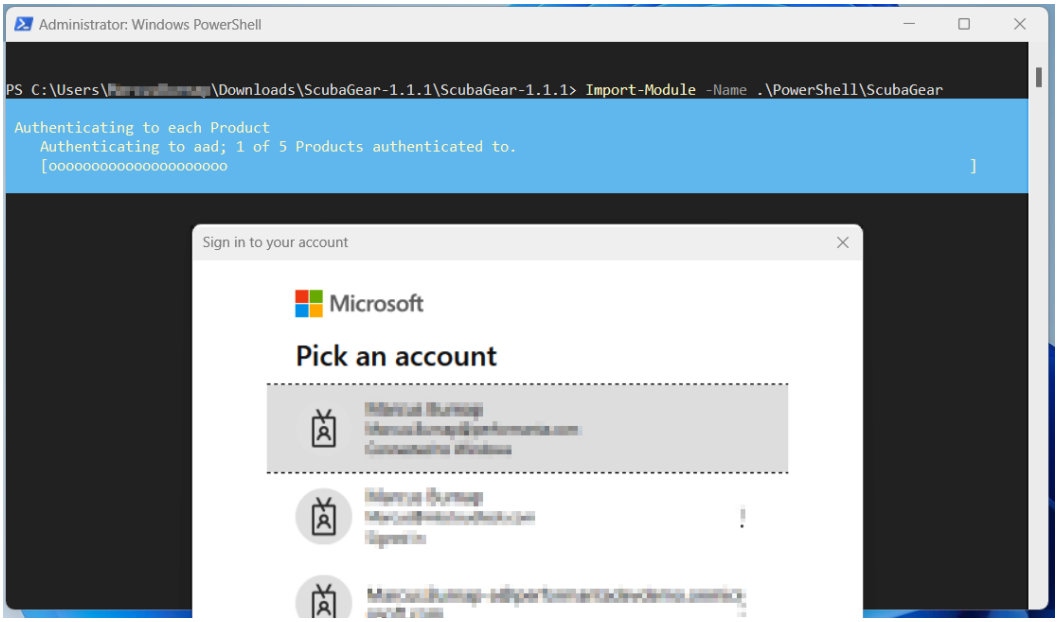


Figure 5.9: Interactive sign-in prompt to complete authentication into the tenant

5. Log in with the relevant details you wish to run the assessment against. Multiple authentications and Microsoft Graph grants will be required.

   Errors based on available licenses and authentication errors will be presented in yellow text like this:

```
WARNING: sharepoint will be omitted from the output because of failed authentication
WARNING: Defender for Office 365 license not available in tenant. Omitting the following commands:
Get-AtpPolicyForO365, Get-ATPProtectionPolicyRule.
```

Figure 5.10: Error based on missing licensing or permissions

Detailed configuration instructions can be found within the `README` file:

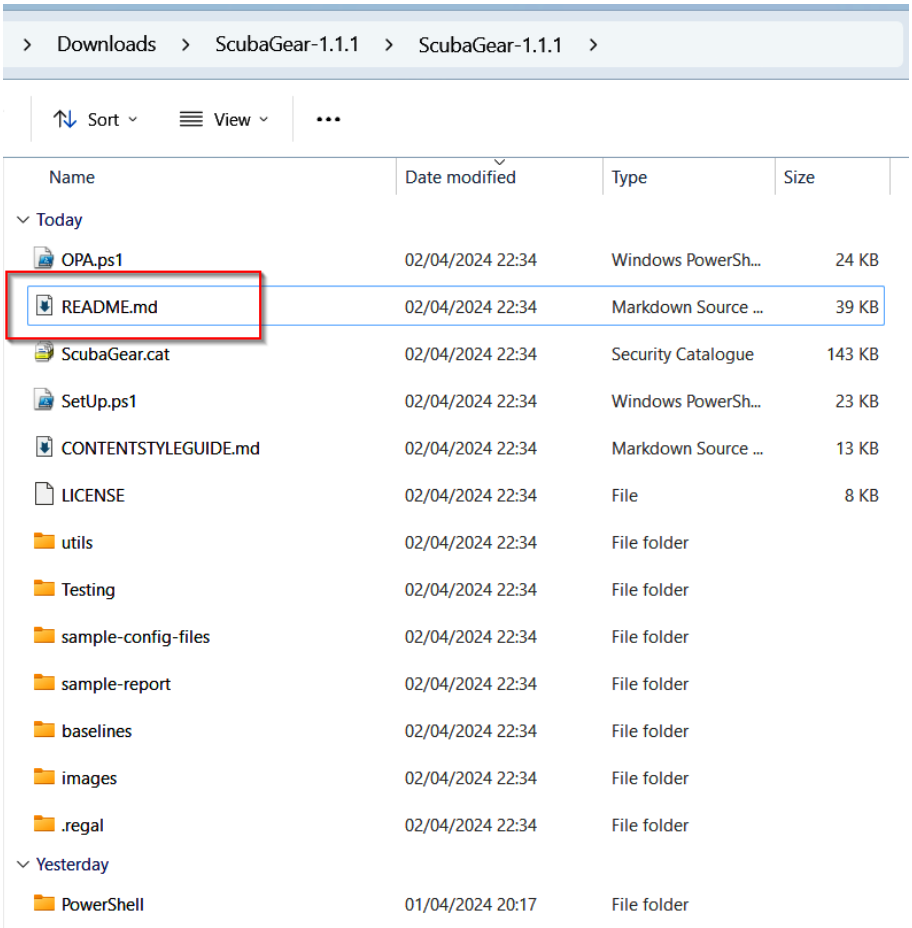


Figure 5.11: README file for ScubaGear configuration instructions

Step 4: ScubaGear output

Once the assessment completes, ScubaGear generates an output in `.html` format, automatically opened in your default web browser.

The output and data contributing to the `.html` presentation are saved in a directory, as shown here:

```
C:/Users/UserName/Downloads/ScubaGear-1.1.1/ScubaGear-1.1.1/
M365BaselineConformance_2024_04_02_23_16_12/BaselineReports.html
```

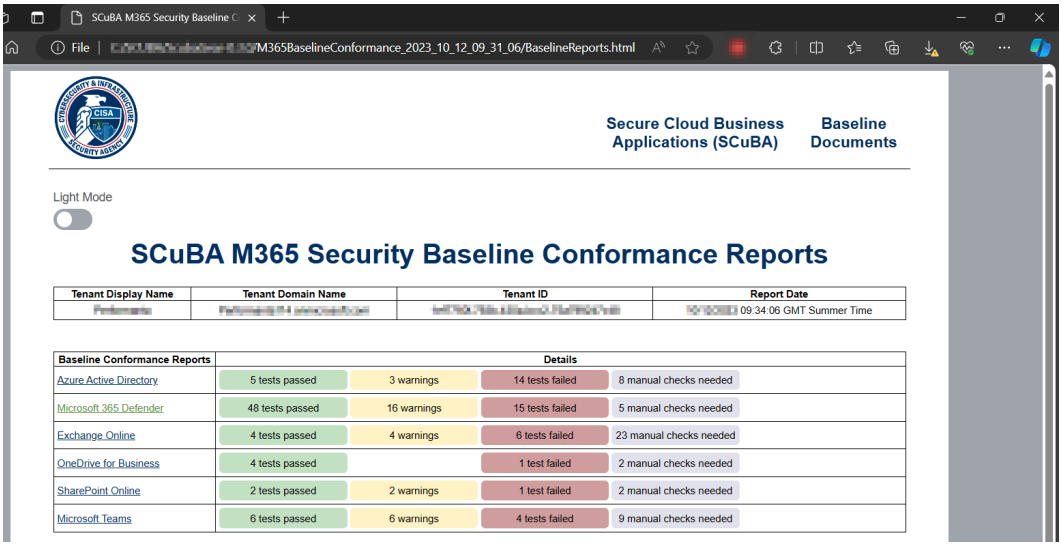


Figure 5.12: ScubaGear final assessment output

This guide consolidates the key steps for setting up and running ScubaGear, including navigating the PowerShell environment, executing the setup script, importing the module, and handling the output for assessment results.

Next, we will learn how to automate tenant monitoring with the Maester tool.

Automating tenant monitoring with Maester and Azure DevOps

In this section, we will build a base setup to run the Maester tool against multiple tenant environments via Azure DevOps. Maester directly addresses Microsoft 365 tenant risks by enabling you to codify your security configuration checks, continuously verifying compliance through automated testing. Leveraging the power of the Pester testing framework, Maester gives you the following advantages:

- Clearly define and version-control your security policy checks as code
- Regularly validate that tenant configurations adhere strictly to these defined policies
- Quickly detect and rectify configuration errors by writing targeted tests that prevent recurrence
- Easily extend your monitoring capabilities through custom tests, adapting quickly to new security requirements and configurations



You can find extensive documentation and an engaged community behind Maester here: `https://maester.dev/`.

Azure DevOps simplifies the daily execution of Maester tests, allowing you to effortlessly schedule these tests and view clear, actionable results directly within the Azure DevOps interface.

Setting up the Maester repository in Azure DevOps

We will now create an Azure DevOps project and configure the tool to run on a scheduled basis. To do this, you will first need to set up the Maester repository in Azure DevOps and then set up the Azure pipeline.

If this is your first time using Azure DevOps, you will first need to create an organization. If you already have an account or choose to use a different platform, such as GitHub with GitHub Actions, please refer to the official site docs: `https://learn.microsoft.com/en-us/azure/devops/organizations/accounts/create-organization?view=azure-devops`.



Microsoft-hosted agent

When using Azure DevOps, you will need a pipeline setup. This will require some free compute power. To enable the free tier, use a Microsoft-hosted agent; for Azure Pipelines, you will need to submit this form: `https://aka.ms/azpipelines-parallelism-request`.

Note that it can take a few days before you can start using the pipeline.

Now that you have created an organization, let us start setting up the repository with the following steps:

1. First, create a new project to host your Maester tests and Azure pipeline.

   For this, navigate to the top-right side of the Azure DevOps project page and select **Azure DevOps | + New project**.

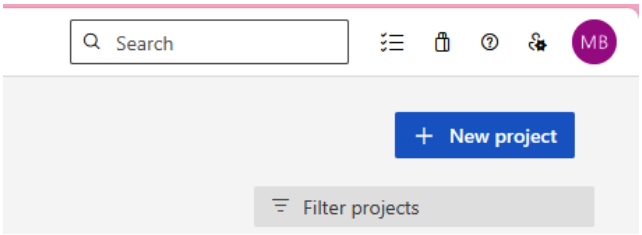


Figure 5.13: Create a new Azure DevOps project

2. Add the project name and description and select the visibility type for your project. Then, click **Create**.

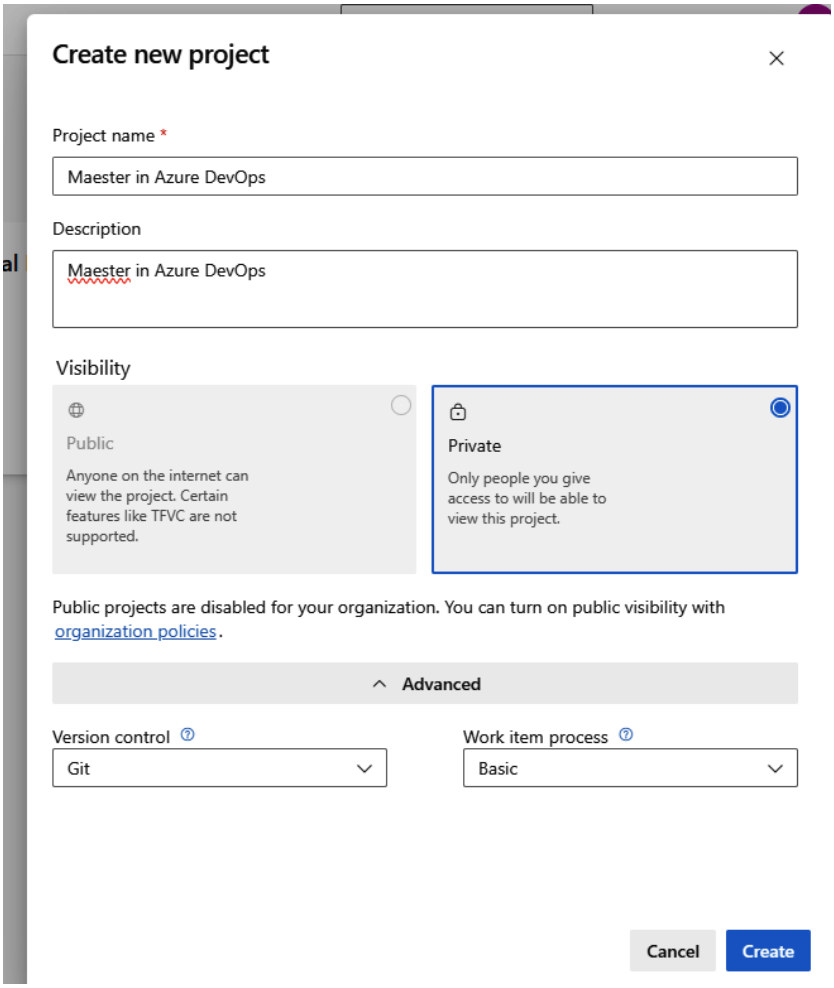


Figure 5.14: Azure DevOps project settings

3. Once you have created the project, import the Maester tests.

   Select **Repos** from the left-hand menu and select **Import repository**.

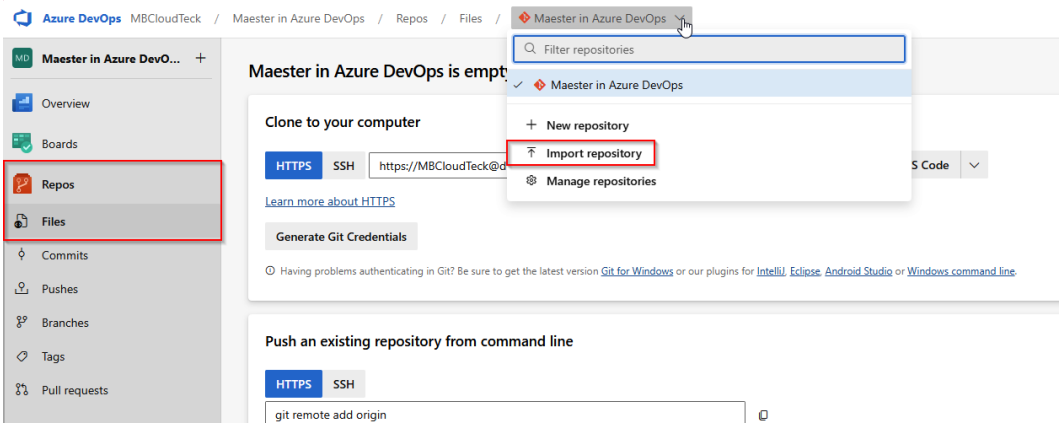


Figure 5.15: Import the Maester repo

Enter the URL of the Maester repository: `https://github.com/maester365/maester-tests`.

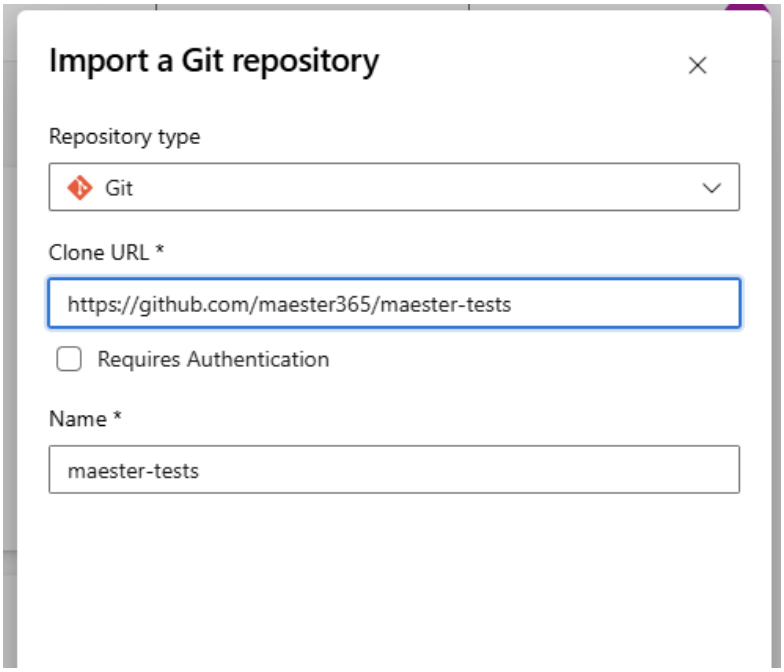


Figure 5.16: Import the Maester GitHub repo

4. At the bottom of the pane, select **Import** to import the repository into your Azure DevOps project. You now have your files imported into your local repo.

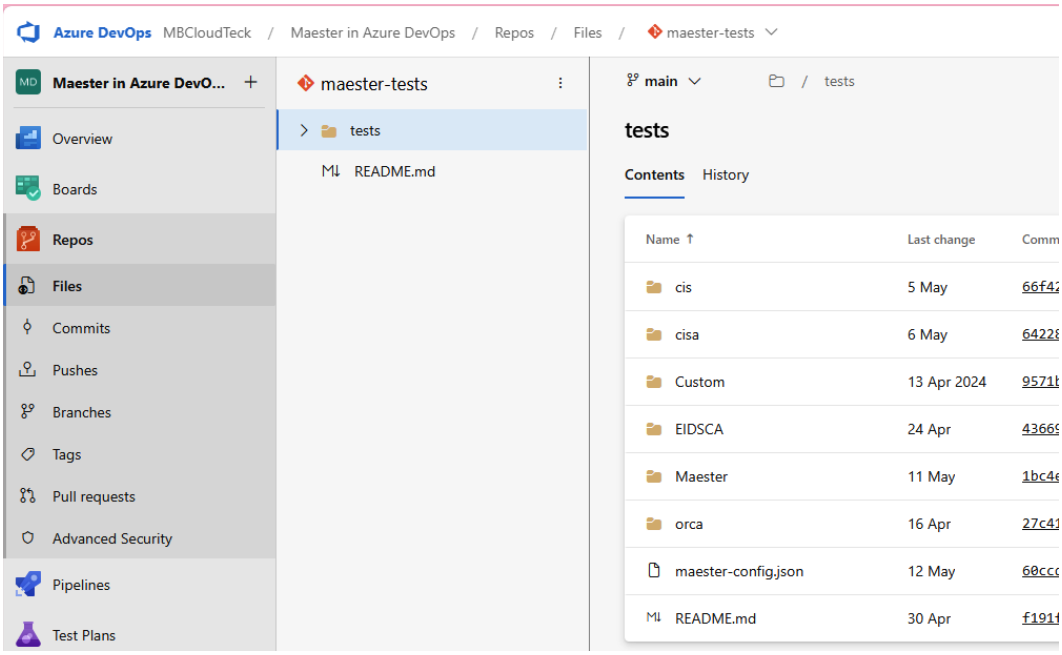


Figure 5.17: Your new local Maester repo

Now that we have the files with tests in the right place, let's learn how to add multiple clients (tenants) into the mix in a structured and manageable way.

Adding a client's location

To be able to run Maester against multiple tenants and keep the repo clean, it's best to create a `clients` folder. This will be used to store the YAML file that will define the variables and client specifications.

To create a new client folder and file, follow these steps:

1. Click on the three dots next to the **Maester** folder and select **create new clients** folder.
2. Then, add a new client file. We recommend that you add a simple name for the client file, as shown in *Figure 5.18*.

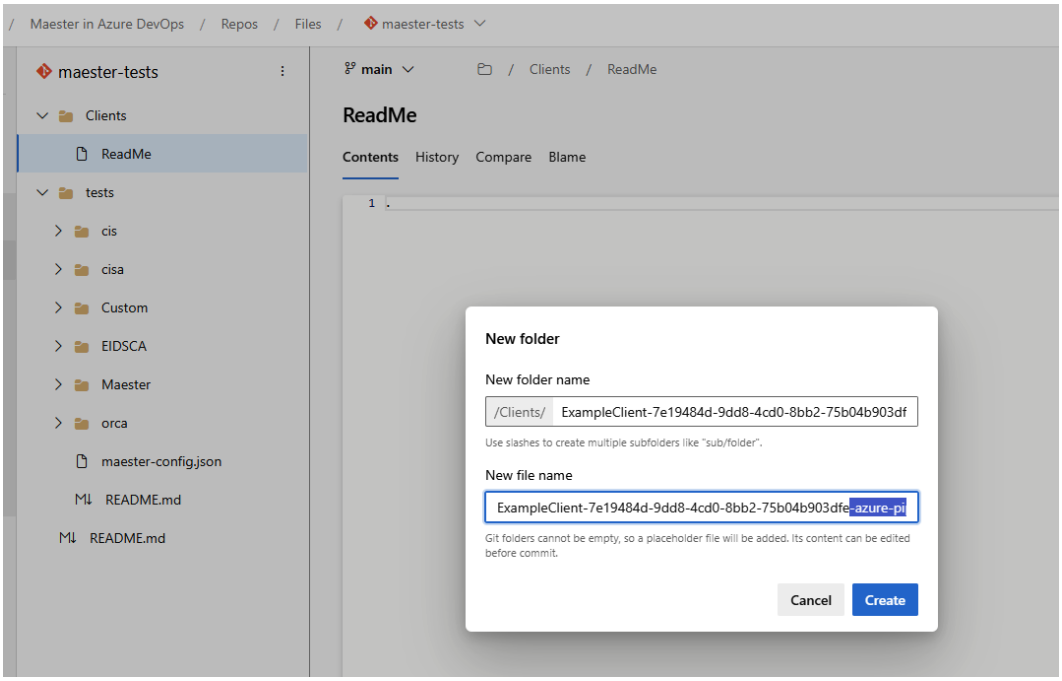


Figure 5.18: Create a new client folder and file

Since you cannot create empty folders, create the file first. When you do, copy and paste the code directly from the demo repository. This ensures the file is saved correctly, and you can avoid the need for edits later.

In case you are new to Azure DevOps, let's break down the full YAML pipeline file shown in *Figure 5.19*.

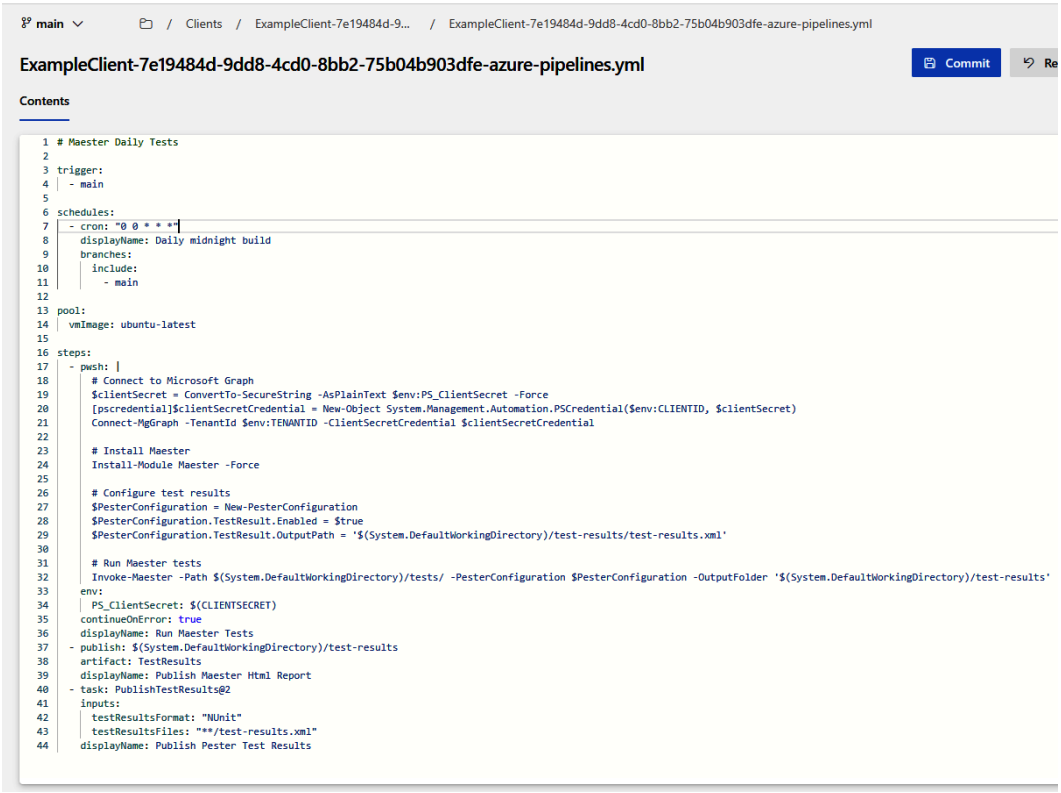


Figure 5.19: An example of a YAML pipeline file

The first section defines the pipeline's continuous integration (CI) trigger. Whenever new code is pushed to the main branch, the pipeline runs automatically. Alternatively, you can configure it to trigger only on specific branches or disable automatic execution entirely by using the –none option.

As shown in *Figure 5.20*, this schedule ensures that tests are run *once per day at midnight (UTC)*, even if no code changes occur.

Contents

```yaml
# Maester Daily Tests

trigger:
  - main

schedules:
  - cron: "0 0 * * *"
    displayName: Daily midnight build
    branches:
      include:
        - main

```

Figure 5.20: Timings of the YAML pipeline file

The `cron` expression (`0 0 * * *`) defines the timing, and the pipeline will only run this schedule for the main branch. This approach is especially effective for detecting drift, expired secrets, or configuration issues that occur over time.

Figure 5.21 shows host pools and authentication.

```yaml
pool:
  vmImage: ubuntu-latest

steps:
  - pwsh: |
      # Connect to Microsoft Graph
      $clientSecret = ConvertTo-SecureString -AsPlainText $env:PS_ClientSecret -Force
      [pscredential]$clientSecretCredential = New-Object System.Management.Automation.PSCredential($env:CLIENTID, $clientSecret)
      Connect-MgGraph -TenantId $env:TENANTID -ClientSecretCredential $clientSecretCredential
```

Figure 5.21: Host pool and authentication of the YAML pipeline file

This shows a PowerShell step that authenticates to Microsoft Graph using a service principal. The script securely converts the client secret into a credential object and then uses the `Connect-MgGraph` cmdlet from the Microsoft Graph PowerShell SDK to establish the connection. The `CLIENTID`, `TENANTID`, and `PS_ClientSecret` values are pulled from environment variables, keeping credentials secure while enabling automated access to Microsoft 365 services.

The next section of the script, as shown in *Figure 5.22*, installs the Maester PowerShell module and prepares the test environment using Pester, a popular testing framework for PowerShell.

```
22
23      # Install Maester
24      Install-Module Maester -Force
25
26      # Configure test results
27      $PesterConfiguration = New-PesterConfiguration
28      $PesterConfiguration.TestResult.Enabled = $true
29      $PesterConfiguration.TestResult.OutputPath = '$(System.DefaultWorkingDirectory)/test-results/test-results.xml'
```

Figure 5.22: Installing the Maester PowerShell module

The `Install-Module` cmdlet ensures that Maester is available for test execution. Then, a new Pester configuration object is created, enabling test result output and specifying the path where the results will be stored. By setting `OutputPath` to `$(System.DefaultWorkingDirectory)/test-results/test-results.xml`, the pipeline ensures that structured test output can be captured and published later in the process.

As shown in *Figure 5.23*, the next portion of the pipeline runs the Maester tests, publishes the results, and integrates them with Azure DevOps test reporting.

```
31      # Run Maester tests
32      Invoke-Maester -Path $(System.DefaultWorkingDirectory)/tests/ -PesterConfiguration $PesterConfiguration -OutputFolder '$(System.DefaultWorkingDirectory)/test-results'
33    env:
34      PS_ClientSecret: $(CLIENTSECRET)
35    continueOnError: true
36    displayName: Run Maester Tests
37  - publish: $(System.DefaultWorkingDirectory)/test-results
38    artifact: TestResults
39    displayName: Publish Maester Html Report
40  - task: PublishTestResults@2
41    inputs:
42      testResultsFormat: "NUnit"
43      testResultsFiles: "**/test-results.xml"
44    displayName: Publish Pester Test Results
```

Figure 5.23: Test reporting

The `Invoke-Maester` command runs all tests located in the `/tests/` directory, using the previously defined Pester configuration and output path. Environment variables such as `PS_ClientSecret` are securely passed into the script, and `continueOnError: true` ensures that even if tests fail, the pipeline will proceed to publish the results.

The next step publishes the `test-results` folder as a downloadable artifact named `TestResults`. Finally, the `PublishTestResults@2` task uploads the structured test output (`test-results.xml`) in `NUnit` format, enabling visibility of pass/fail status and trends directly within the Azure DevOps UI.

We have now triggered the pipeline by changing the main branch and a midnight CRON schedule. We also used a Linux-based build agent (`ubuntu-latest`) to authenticate Microsoft Graph via service principal credentials. Then, we installed the Maester module on the pipeline, configured Pester for structured XML output, and ran the test suite located in the project's `/tests/` directory. You can now find the results in a defined folder, published as a downloadable artifact.

Setting up the Azure pipeline

We now need to create a pipeline to run the imported test against the tenants. This will require the correct authentication for the pipeline to connect with. There are several ways to authenticate with Microsoft Entra in Azure DevOps. Here, we'll use a service principal with a client secret, enabling access to both the Exchange and Azure modules.



The Azure RBAC permissions are required for running validation tests on Azure configurations, including tests aligned with CISA standards.

Similarly, Exchange Online permissions are needed to conduct validation tests on Exchange Online configurations, also aligned with CISA standards.

To enable secure authentication for each client tenant, the next step is to create an Entra ID application registration that the Azure pipeline will use.

Creating an Entra ID application registration

An Entra ID application registration should be created in your client tenant or the tenant you wish to run the Maester tests against. You can authenticate this from your central Azure DevOps location to each individual client via this service principal in the client tenant. To do this, follow these steps:

1. Open the Entra admin center and navigate to **Identity | Applications | App registrations | New registration**.

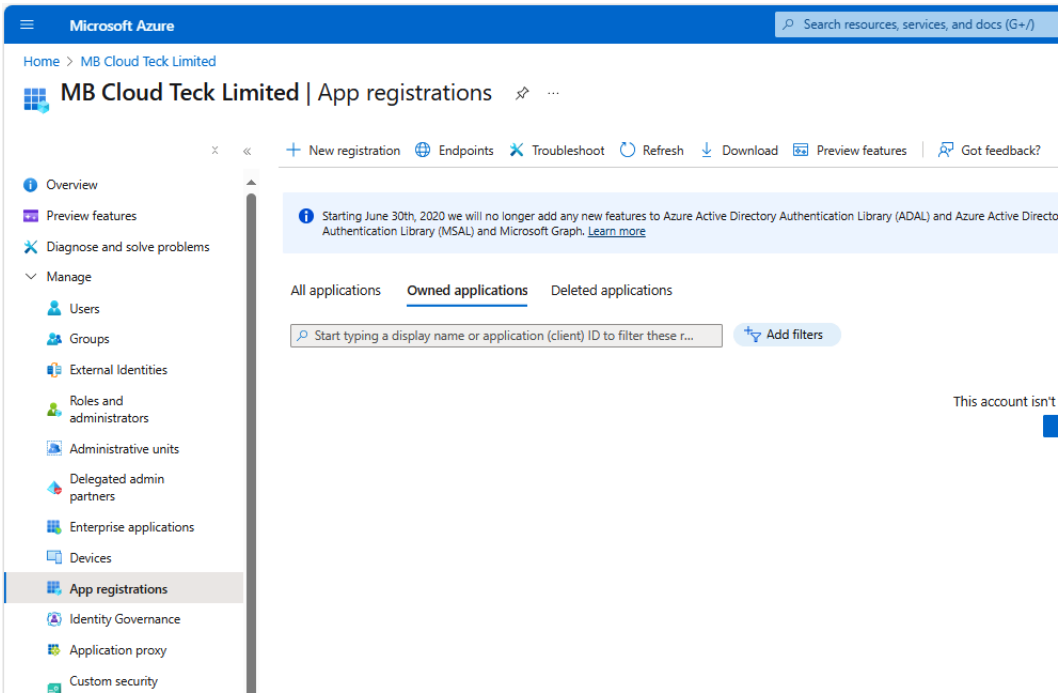


Figure 5.24: Create an app registration for Maester 1

2. Enter a name for the application (e.g., `Maester DevOps Account`) and select supported account types and redirect URI platform you prefer. Then, click on **Register**.

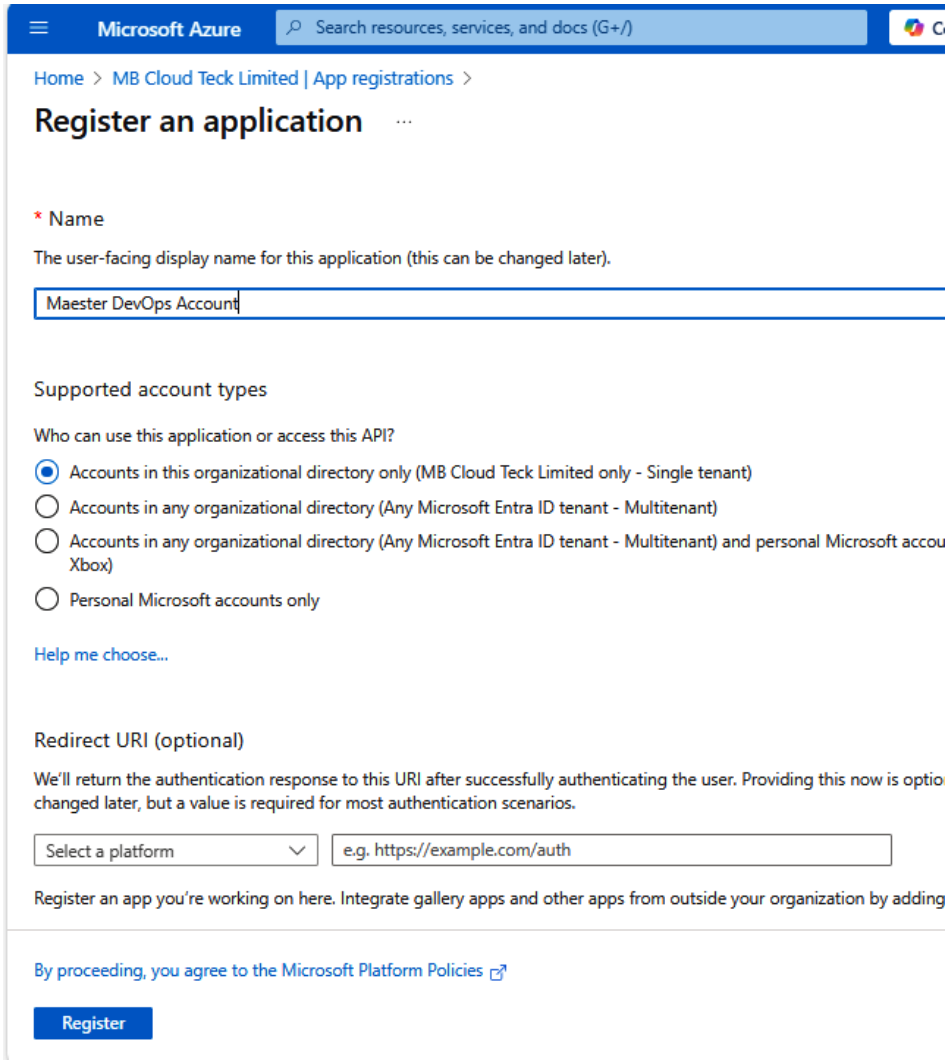


Figure 5.25: Create an App Registration for Maester 2

3. Next, we need to grant permissions to Microsoft Graph. Open the application you created in the previous step. Then, select **API permissions | Add a permission**.

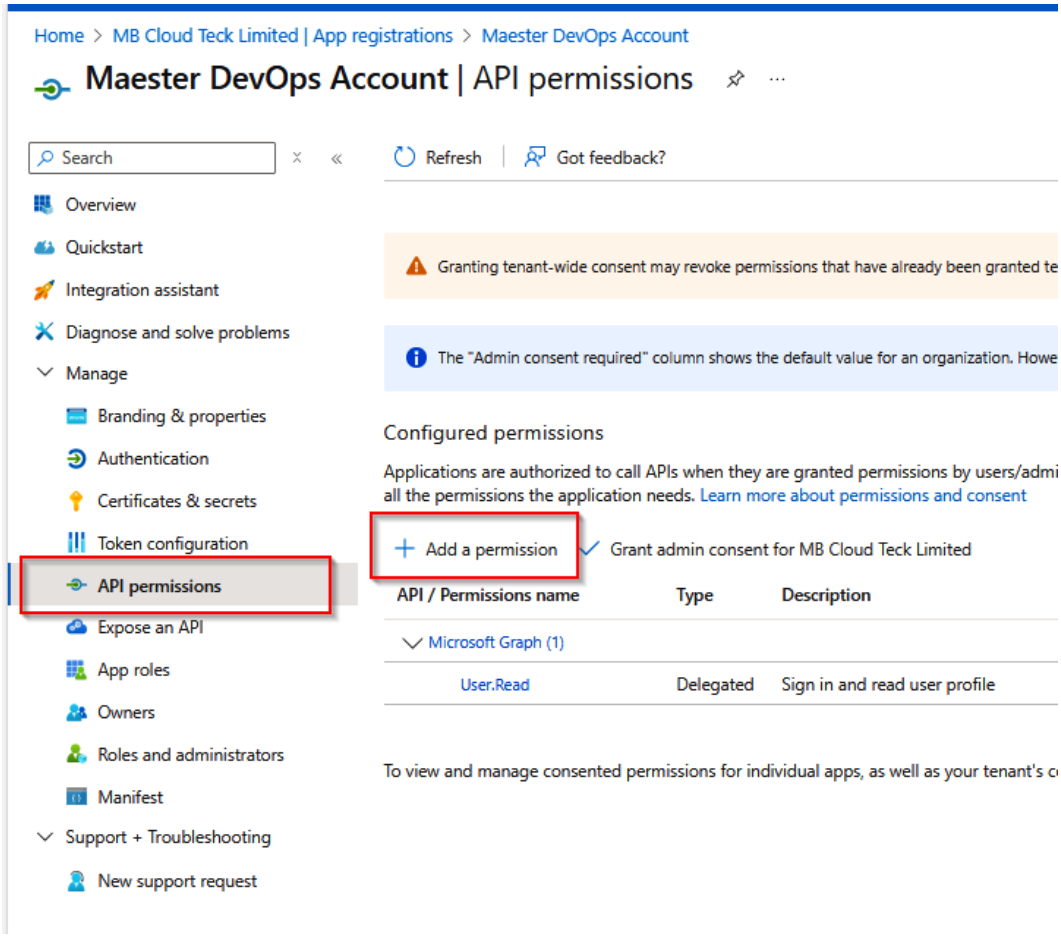


Figure 5.26: Granting API access for Maester

4. Then, select **Request API permissions**. This will take you to the **Microsoft APIs** tab. Click on **Microsoft Graph**.

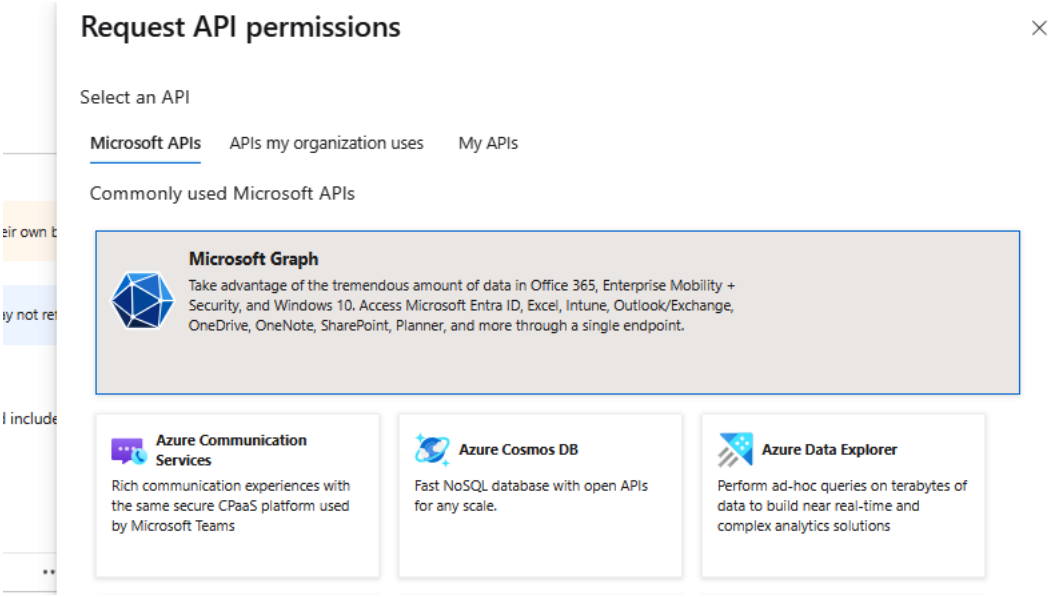


Figure 5.27: Selecting Microsoft Graph API access for Maester

5. Now, select **Microsoft Graph | Application permissions**. This will take you to a pop-up window where you select app permissions.

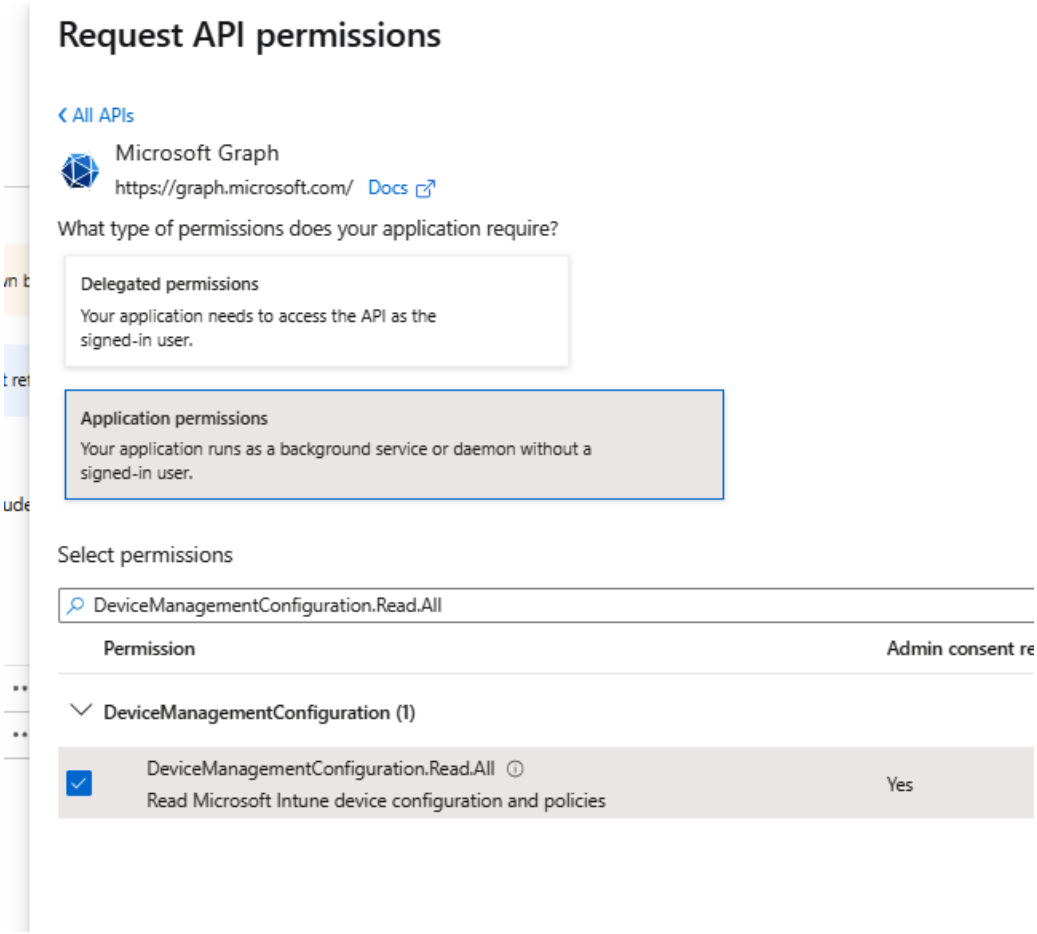


Figure 5.28: Selecting the required application access for Maester

6. Search for each of the following permissions and check the box next to each permission:

    - **DeviceManagementConfiguration.Read.All**
    - **DeviceManagementManagedDevices.Read.All**
    - **Directory.Read.All**
    - **DirectoryRecommendations.Read.All**
    - **IdentityRiskEvent.Read.All**
    - **Policy.Read.All**
    - **Policy.Read.ConditionalAccess**

- **PrivilegedAccess.Read.AzureAD**
- **Reports.Read.All**
- **RoleEligibilitySchedule.Read.Directory**
- **RoleManagement.Read.All**
- **SharePointTenantSettings.Read.All**
- **UserAuthenticationMethod.Read.All**
- **RoleEligibilitySchedule.ReadWrite.Directory**

7. Now, click on **Add permissions**, then **Grant admin consent for [your organization]**, and then click **Yes** to confirm.

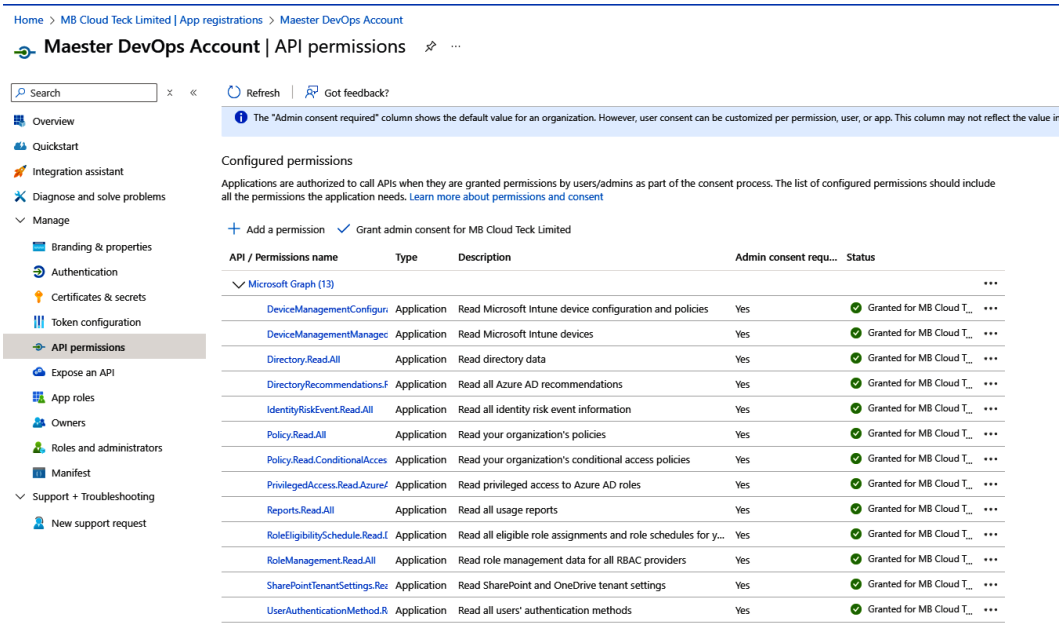


Figure 5.29: Completed Graph API access for Maester

Now that we have completed the Entra ID app registration, let's briefly look at how to grant permissions to Exchange Online.

Granting permissions to Exchange Online (optional but recommended)

The Exchange Online role-based access control implementation utilizes service-specific roles that apply to an application. The following configuration enables an authorization chain to the app registration you created in the previous steps:

1. Open the application you created in the previous process.
2. Navigate to **API permissions | Add a permission | APIs that my organization uses**. Here, search for `Office 365 Exchange Online` and then select **Application permissions**.
3. Then, search for `Exchange.ManageAsApp`. Now, select **Add permissions | Grant admin consent for [your organization] | Yes**.
4. Connect to the Exchange Online management tools and add the following script to set the appropriate permissions:

```
New-ServicePrincipal -AppId <Application ID> -ObjectId <Object ID>
-DisplayName <Name>
New-ManagementRoleAssignment -Role "View-Only Configuration" -App
<DisplayName from previous command>
```

Granting permissions to Azure (optional)

The Azure role-based access control implementation utilizes **Uniform Resource Names (URNs)** with a `/` separator for hierarchical scoping. There are resources within such root scopes that Microsoft retains strict control over by limiting supported interactions. As a global administrator, you can elevate access: (`https://learn.microsoft.com/en-us/azure/role-based-access-control/elevate-access-global-admin?tabs=powershell`) to become authorized for these limited interactions.



The Exchange Online permissions are necessary to support tests that validate Exchange Online configurations, such as the CISA tests: `https://maester.dev/docs/tests/cisa/exo`.

The Azure role-based access control permissions are necessary to support tests that validate Azure configurations: `https://maester.dev/docs/installation#installing-azure-and-exchange-online-modules`, such as the CISA tests: `https://maester.dev/docs/tests/cisa/entra#:~:text=Test%2DMtCisaDiagnosticSettings`.

As a Global Administrator, you can run the following PowerShell script to securely configure reader permissions at the tenant and root scope levels:

```
$servicePrincipal = "<Object ID of the Entra App>"
$subscription = "<Subscription ID>"
Install-Module Az.Accounts -Force
Install-Module Az.Resources -Force
Connect-AzAccount
#Elevate to root scope access
$elevateAccess = Invoke-AzRestMethod -Path "/providers/Microsoft.
Authorization/elevateAccess?api-version=2015-07-01" -Method POST

#Assign permissions to Enterprise App
New-AzRoleAssignment -ObjectId $servicePrincipal -Scope "/"
-RoleDefinitionName "Reader" -ObjectType "ServicePrincipal"
New-AzRoleAssignment -ObjectId $servicePrincipal -Scope "/providers/
Microsoft.aadiam" -RoleDefinitionName "Reader" -ObjectType
"ServicePrincipal"

#Remove root scope access
$assignment = Get-AzRoleAssignment -RoleDefinitionId 18d7d88d-d35e-
4fb5-a5c3-7773c20a72d9|?{$_.Scope -eq "/" -and $_.SignInName -eq (Get-
AzContext).Account.Id}
$deleteAssignment = Invoke-AzRestMethod -Path "$($assignment.
RoleAssignmentId)?api-version=2018-07-01" -Method DELETE
```

This script will help you in the following areas:

- Identify the service principal object ID that will be authorized as a reader (enterprise app object ID)
- Install the necessary `Az` module and prompt for connection
- Elevate your account access to the root scope
- Create a role assignment for reader access over the root scope
- Create a role assignment for reader access over Entra ID (i.e., **aadiam provider**: `https://learn.microsoft.com/en-us/azure/role-based-access-control/permissions/identity#microsoftaadiam`)

- Identify the role assignment authorizing your account access to the root scope
- Delete the root scope role assignment for your account

With the required permissions now in place and access validated, the next step is to enable secure, programmatic authentication. To do this, you need to create a client secret for the service principal. This allows applications and automation to authenticate without user interaction.

Creating a client secret

Navigate back to the **Create application registration** section and follow the instructions to create a client secret:

1. Click on **Certificates & secrets | Client secrets | New client secret**.
2. Enter a description for the secret (e.g., `Maester DevOps Secret`) and select **Add**.
3. Copy the value of the secret. We will use this value in the Azure pipeline:

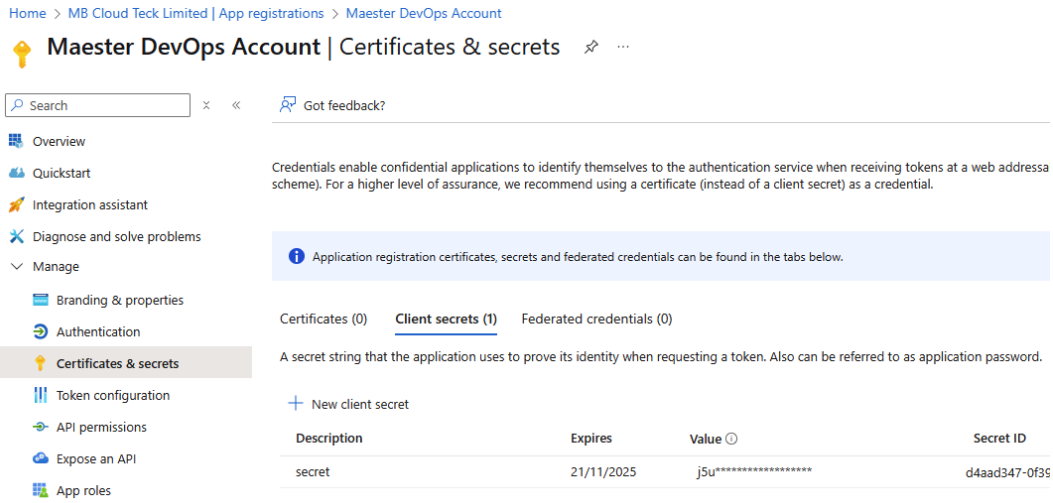


Figure 5.30: Creating an app registration secret

With the client secret created and securely stored, you're now ready to integrate it into your automation workflow.

Creating an Azure pipeline

Follow these steps to configure an Azure DevOps pipeline. You'll use the secret copied in the previous process for authenticating and interacting with Microsoft services:

1. Open your Azure DevOps project. Select **Pipelines | New pipeline**. You will see the new pipeline screen.
2. Select **Azure Repos Git** as the location of your code:

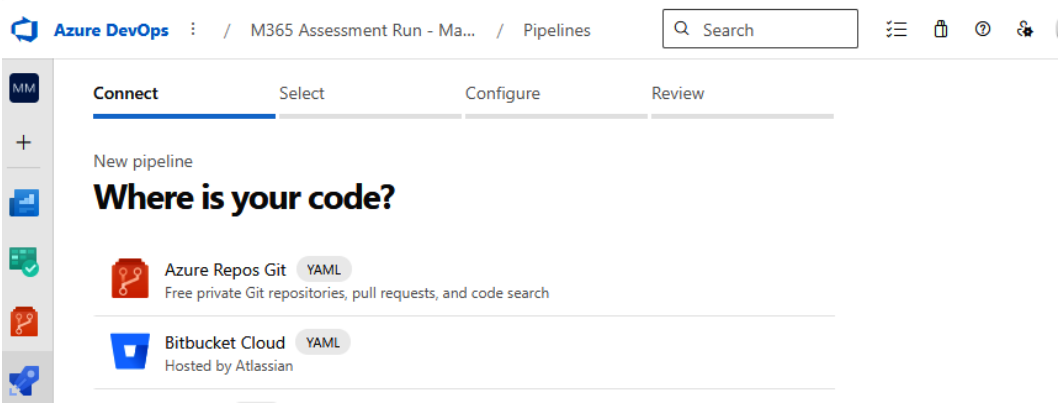


Figure 5.31: Selecting Azure Repos Git

3. Select the repository where you imported the Maester tests. Then, select **Starter pipeline**.

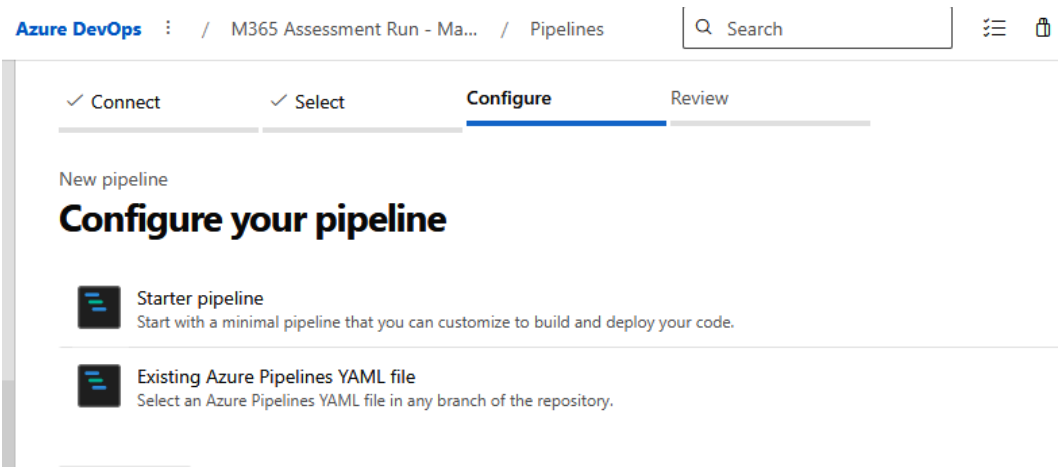


Figure 5.32: Creating a new starter DevOps pipeline

4. Select the **Variables** button to open the variables editor and add the following variables:

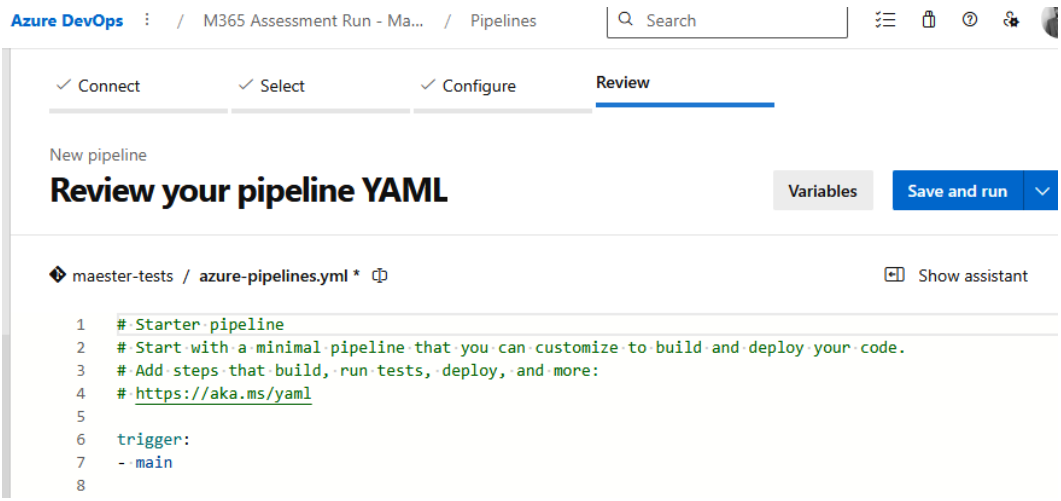


Figure 5.33: Adding variables to the Azure DevOps pipeline

5. In the Entra portal, open the application registration you created earlier and copy the following values from the **Overview** page:

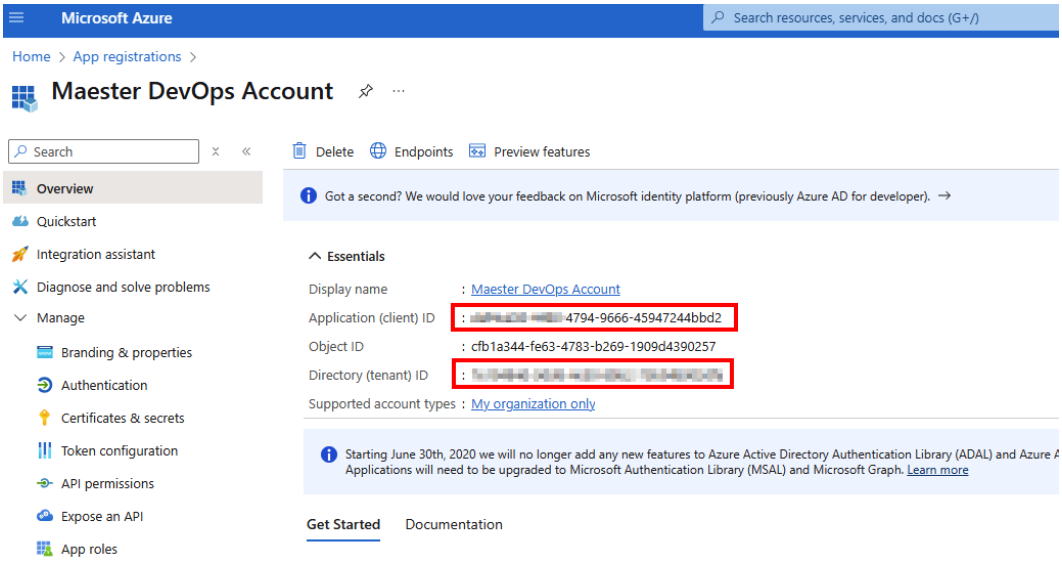


Figure 5.34: Retrieving the service principal details for the Azure DevOps pipeline

6. Now, use these values in the pipeline:
    - **Name**: `TENANTID`; **Value**: The directory (tenant) ID of the Entra tenant
    - **Name**: `CLIENTID`; **Value**: The application (client) ID of the Entra application you created
    - **Name**: `CLIENTSECRET`; **Value**: The client secret you copied in the previous step
    - Important: Tick the **Keep this value secret** checkbox

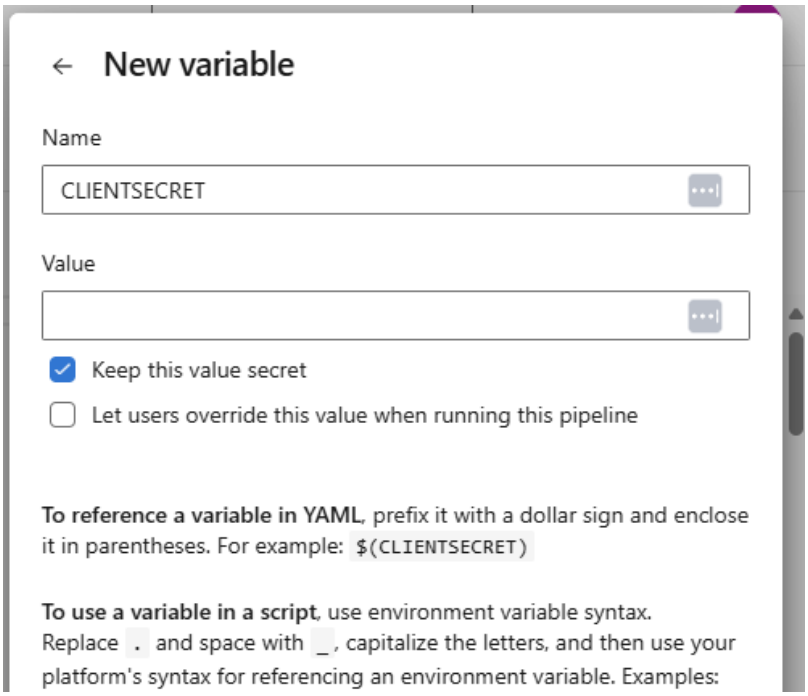


Figure 5.35: Adding the client secret securely

You should be left with the values shown in *Figure 5.36*:

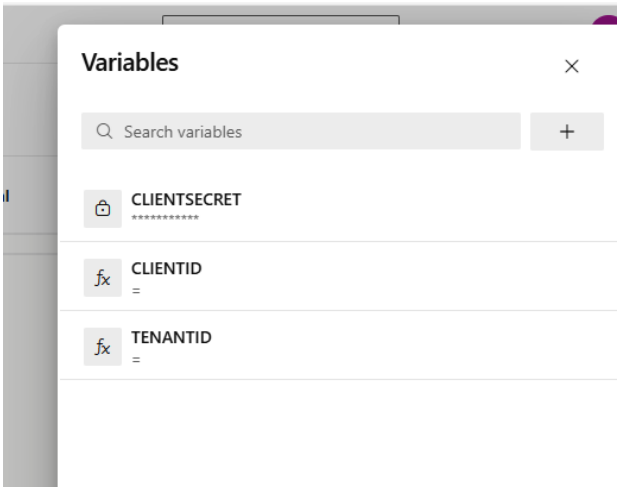


Figure 5.36: Final variables for the Azure DevOps pipeline

7. Replace the content of the `azure-pipelines.yml` file with the following code:

```yaml
# Maester Daily Tests

trigger:
  - main

schedules:
  - cron: "0 0 * * *"
    displayName: Daily midnight build
    branches:
      include:
        - main

pool:
  vmImage: ubuntu-latest

steps:
  - pwsh: |
      # Connect to Microsoft Graph
      $clientSecret = ConvertTo-SecureString -AsPlainText $env:PS_ClientSecret -Force
      [pscredential]$clientSecretCredential = New-Object System.Management.Automation.PSCredential($env:CLIENTID, $clientSecret)
      Connect-MgGraph -TenantId $env:TENANTID -ClientSecretCredential $clientSecretCredential

      # Install Maester
      Install-Module Maester -Force

      # Configure test results
      $PesterConfiguration = New-PesterConfiguration
      $PesterConfiguration.TestResult.Enabled = $true
      $PesterConfiguration.TestResult.OutputPath = '$(System.DefaultWorkingDirectory)/test-results/test-results.xml'

      # Run Maester tests
      Invoke-Maester -Path $(System.DefaultWorkingDirectory)/tests/ -PesterConfiguration $PesterConfiguration -OutputFolder '$(System.DefaultWorkingDirectory)/test-results
    env:
      PS_ClientSecret: $(CLIENTSECRET)
    continueOnError: true
    displayName: Run Maester Tests
  - publish: $(System.DefaultWorkingDirectory)/test-results
    artifact: TestResults
    displayName: Publish Maester Html Report
  - task: PublishTestResults@2
    inputs:
      testResultsFormat: "NUnit"
      testResultsFiles: "**/test-results.xml"
    displayName: Publish Pester Test Results
```

Figure 5.37: Replace the content of the azure-pipelines.yml file

8. Select **Validate and save | Save**.
9. Select **Run** to run the pipeline. You will see what is shown in the following screenshot once it's run successfully.

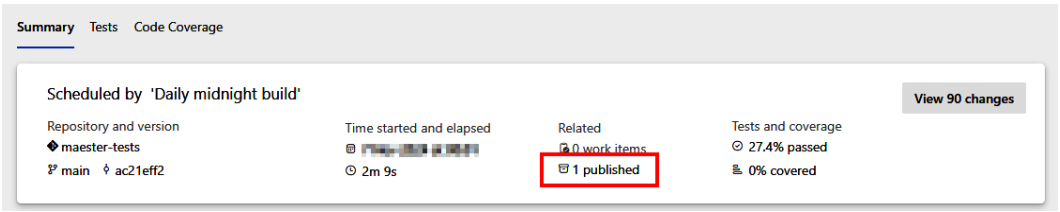


Figure 5.38: Azure DevOps pipeline successfully run – retrieve output

10. Select **Job** to view the test results, as shown in *Figure 5.39*:

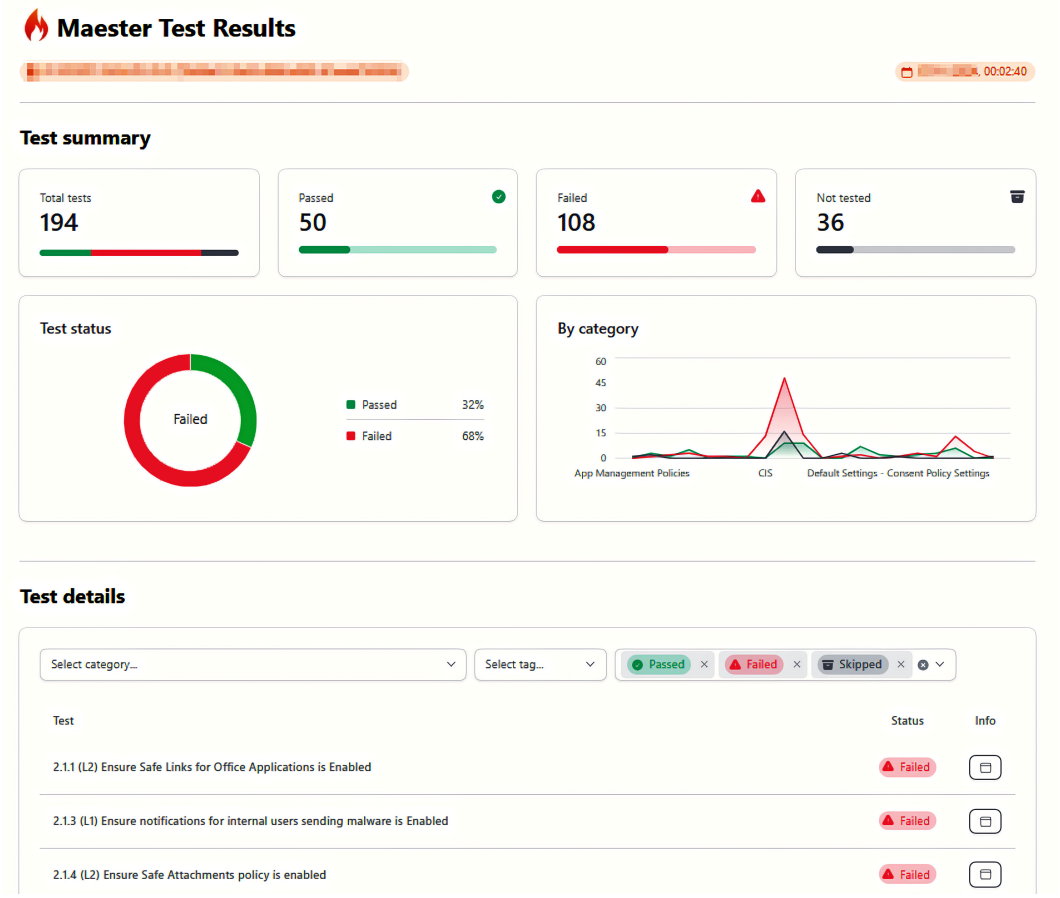


Figure 5.39: Example of the Maester HTML output page

With the Azure DevOps pipeline now configured and securely authenticated, we will now shift our focus to assessing and strengthening our security posture. The next section introduces the Zero Trust Assessment Tool and explains how it can be used to evaluate alignment with zero trust principles and identify areas for improvement.

Zero Trust Assessment Tool guide

The Zero Trust Assessment Tool is a PowerShell cmdlet designed to evaluate your Microsoft 365 tenant's security baseline. It identifies configuration gaps and provides actionable recommendations to help you adopt a zero trust architecture and strengthen your overall security posture.

The tool uses the Microsoft Graph API to assess the current state of your tenant's security configuration and returns a detailed report in Excel (`.xlsx`) format. This will help you to review and align your tenant settings with zero trust principles.



No personal or sensitive tenant data is collected. The only telemetry gathered (if enabled) is the Entra ID tenant ID (GUID).

Prerequisites

Before you can run the assessment, you will need PowerShell 7.0 as well as an admin account with the necessary permissions to execute the assessment.

To install PowerShell 7 using WinGet, follow these steps:

1. Search for the latest version of PowerShell: `winget search Microsoft.PowerShell`.
2. Install PowerShell or PowerShell Preview using the `id` parameter: `winget install --id Microsoft.PowerShell --source winget`.

It should look as shown in *Figure 5.40*:

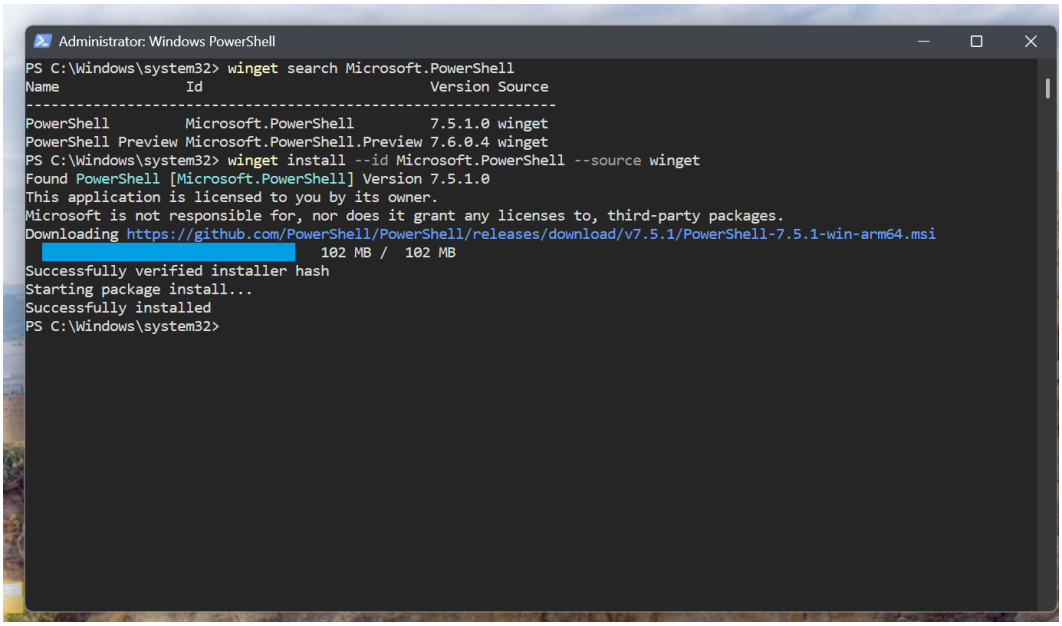


Figure 5.40: Starting PowerShell for the zero trust assessment

You will also need an admin account. Use a non-guest account with admin privileges to run the tool, such as `<username>@contoso.onmicrosoft.com`.

The tool also requests additional read-only permissions that require application administrator consent. These include the following:

- `Agreement.Read.All`
- `CrossTenantInformation.ReadBasic.All`
- `Directory.Read.All`
- `Policy.Read.All`
- `User.Read`
- `DeviceManagementServiceConfig.Read.All`
- `DeviceManagementConfiguration.Read.All`
- `DeviceManagementRBAC.Read.All`
- `DeviceManagementApps.Read.All`

- `RoleAssignmentSchedule.Read.Directory`
- `RoleEligibilitySchedule.Read.Directory`
- `PrivilegedEligibilitySchedule.Read.AzureADGroup`



No tenant data is stored, and sessions are revoked upon sign-out.

Let's now learn how to install and run the tool.

First-time module installation

To install the module, complete the following steps:

1. Launch PowerShell 7 as an administrator.
2. Run `Install-Module ZeroTrustAssessment` to install the module.

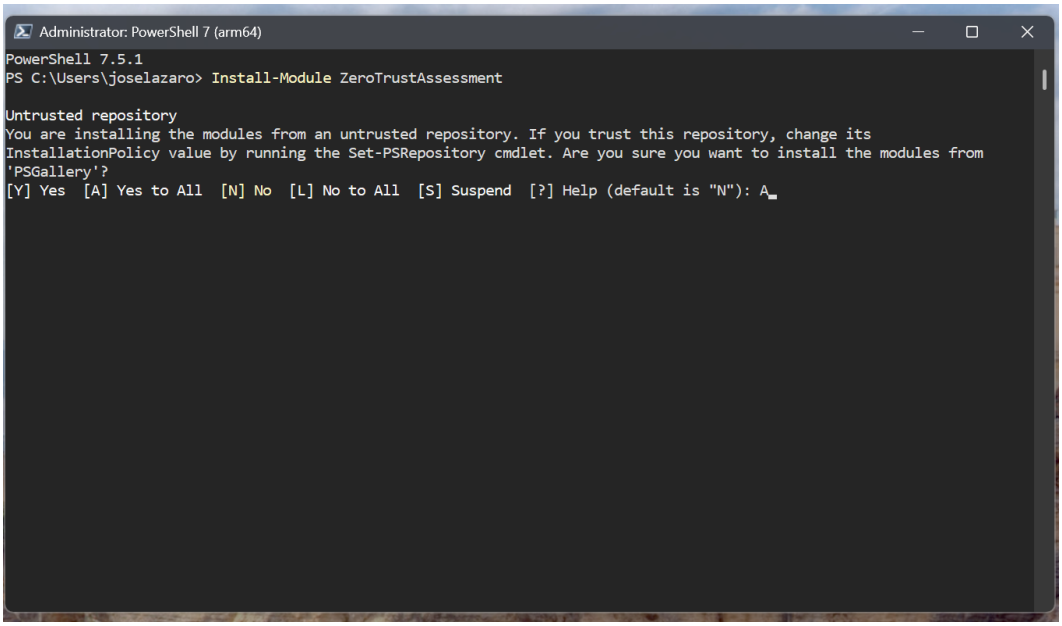


Figure 5.41: Installing the module

3. Then, run `Invoke-ZTAssessment`.

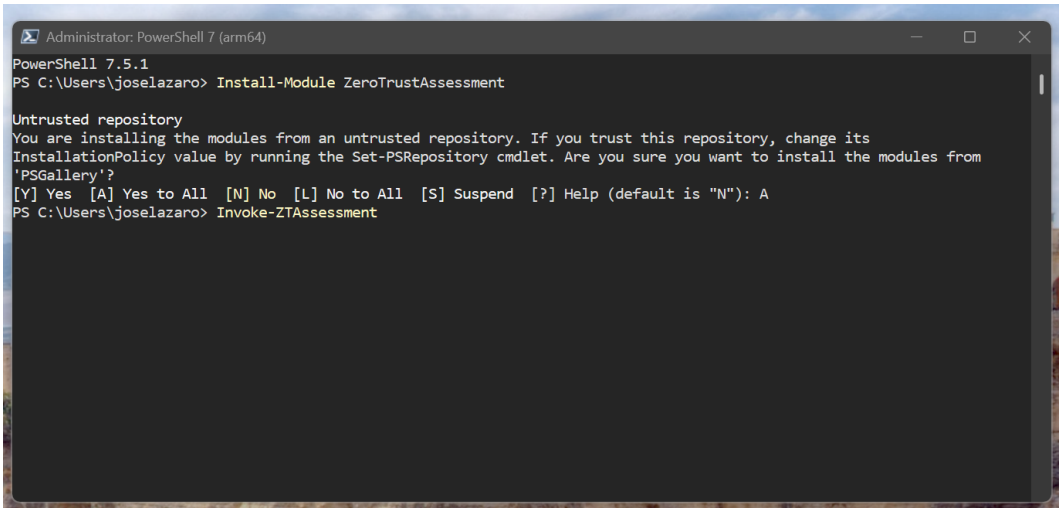


Figure 5.42: Invoking the module

4. Sign in using an admin account.

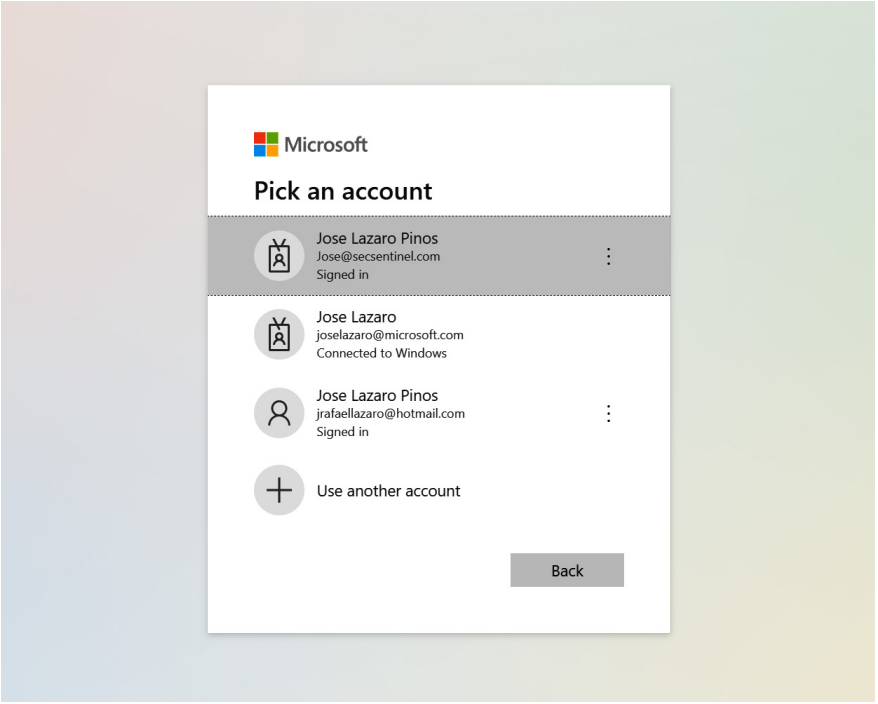


Figure 5.43: Authorizing with an account

5. Click **Accept** to consent to all required permissions on behalf of your organization.

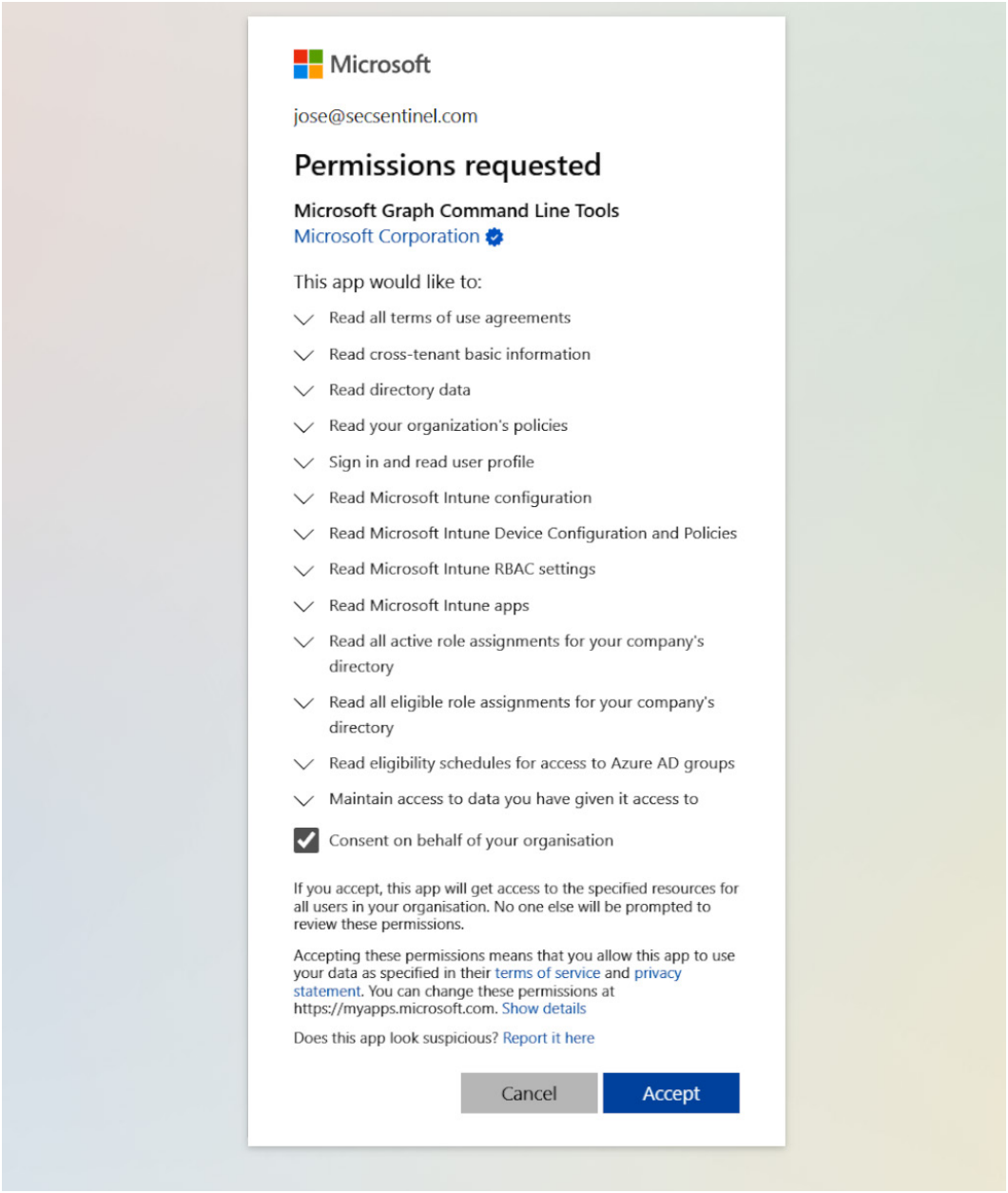


Figure 5.44: Consenting to the required Graph permissions

6. A screen with the confirmation text **Authentication complete. You can return to the application. Feel free to close this browser tab.** will appear.
7. Then, you'll receive the output file location (e.g., `C:\Users\<yourname>`).

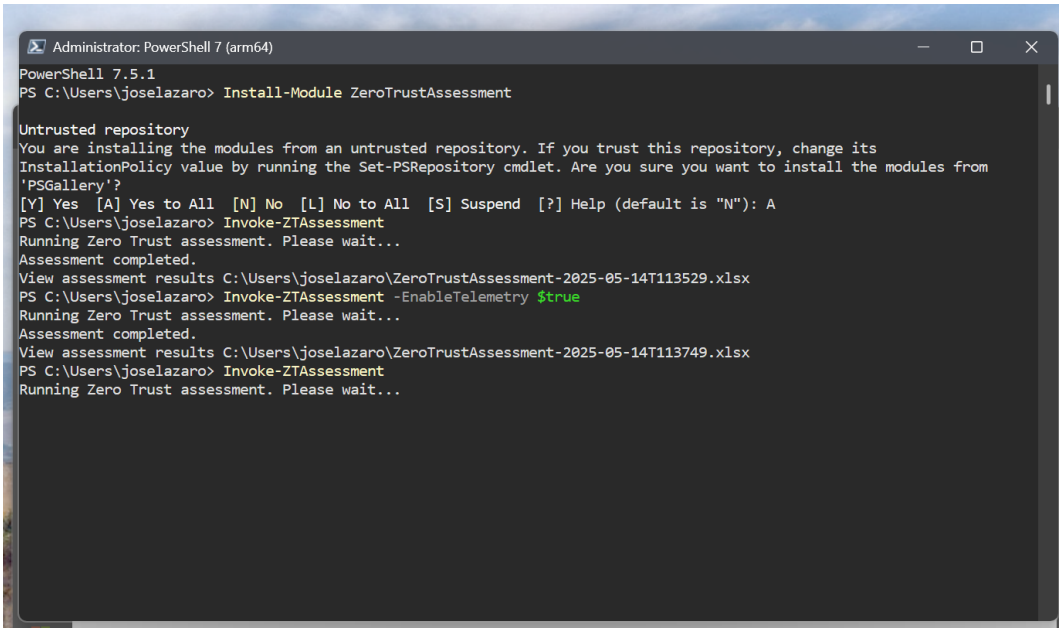


Figure 5.45: Completing the assessment and exporting

8. The output will be a `.xlsx` file. Open it to review the tabs and configurations.

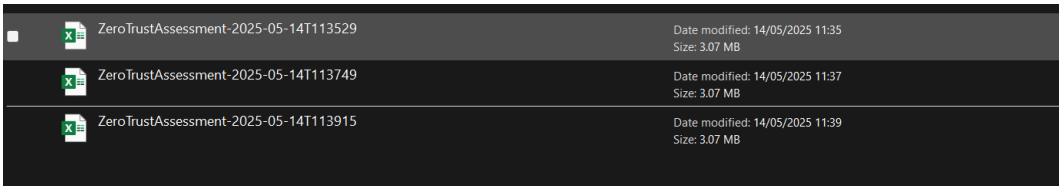


Figure 5.46: Zero trust assessment export

9. The document should look as shown in *Figure 5.47*, containing the assessment results:

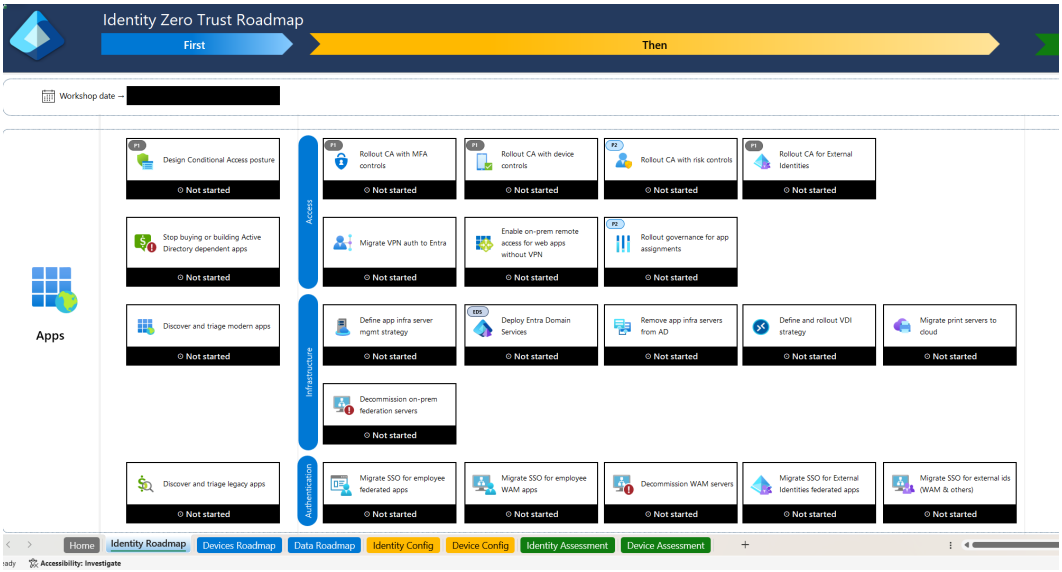


Figure 5.47: Zero trust assessment roadmap



For subsequent runs of the assessment, use `Import-Module` instead:

```
Import-Module ZeroTrustAssessment
Invoke-ZTAssessment
```

You have now installed the module, and it is ready for use. Now we will look at some optional telemetry settings that will allow you to control what usage and diagnostic data is collected during operation.

Telemetry settings (optional)

By default, telemetry is *enabled* to collect only the Entra ID tenant ID (GUID). No other data is stored or transmitted.

You can control this using the `-EnableTelemetry` switch:

```
Invoke-ZTAssessment -EnableTelemetry $true   # Default
Invoke-ZTAssessment -EnableTelemetry $false  # Disable telemetry
```



To access the workshop guidance, go to `https://aka.ms/ztworkshop`. There, you'll find all the necessary documentation from the main contributors of the amazing project!

Feeding Security Copilot with real context: Maester test results

With baseline testing in place, you can move beyond reactive alerting and toward context-driven security recommendations powered by AI. One emerging opportunity is to use the output from tools such as Maester to inform Microsoft Security Copilot, Microsoft's AI assistant for security teams. There is no native way to do this yet; however, something such as a logic app flow could easily accomplish the task.

Security Copilot is most effective when it has relevant, timely, and detailed data and context. Maester focuses on delivering that. By running Maester tests on a scheduled basis and exporting structured results in XML or JSON format, you can create a high-fidelity signal that reflects your tenant's configuration health, policy enforcement, and potential misalignments. These results can be ingested by Copilot as part of Copilot prompts, through integrations with Sentinel workbooks or hunting queries, or by feeding alerts and insights into Defender XDR's advanced hunting context.

For example, Security Copilot can start its analysis with reports such as the following:

- *MFA is not enforced on 28% of guest accounts*
- *Teams external sharing is enabled in three policy exceptions*
- *Your Conditional Access rules are misaligned with your zero trust baseline*

These data points make investigations more accurate and help Copilot recommend more focused and streamlined actions rooted in real-world tenant telemetry.



Watch this space: Maester to Security Copilot

In 2026, we can expect community-driven contributions that map Maester output formats directly into Security Copilot-compatible prompts and structured inputs. These integrations aim to bridge DevSecOps testing pipelines with AI-assisted security operations, unlocking new layers of automation, insight, and proactive response.

As you move forward into deploying Microsoft Sentinel, consider how these early-stage assessments can become long-term signal generators for your AI-powered security ecosystem.

Summary

Before deploying Microsoft Sentinel, validating the security posture of your Microsoft 365 tenant is essential. This chapter introduced you to a practical, tools-driven approach to assess and strengthen your tenant using industry-aligned baselines and automated testing frameworks. We began with ScubaGear, a free tool from CISA that evaluates your Microsoft 365 environment against the SCuBA baseline. With its HTML-based reports, ScubaGear offers immediate visibility into misconfigurations, license dependencies, and policy misalignments.

Next, we explored Maester, a DevSecOps-friendly, PowerShell-based testing framework designed to codify your tenant security policies. By integrating with CI/CD platforms such as Azure DevOps, Maester enables continuous validation across environments, supporting repeatability, automation, and policy as code. Finally, the Zero Trust Assessment Tool provided you with a strategic view into your current zero trust maturity. By running lightweight, permission-aware scans, this tool helps you prioritize and sequence your tenant's progression across identity, device, and access controls.

With your Microsoft 365 tenant assessed and baseline configurations in place, the next step is preparing the ground for Microsoft Sentinel itself. In *Chapter 6*, we will focus on Sentinel's foundational architecture, starting with how to organize your Azure environment using management groups, subscriptions, and Log Analytics workspaces. You'll also learn how to implement structured access control using Microsoft Entra ID, PIM, and access packages, and how to extend that model to multitenant environments using Azure Lighthouse and GDAP.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



6

Foundational Architecture and Secure Access for Sentinel Deployments for MSSPs

This chapter sets the stage for a successful Microsoft Sentinel deployment by focusing on the architectural and access control foundations that underpin a secure and scalable environment. Whether you're deploying Sentinel in a new Azure tenant or managing a complex multitenant **Managed Security Service Provider (MSSP)** scenario, this chapter provides the building blocks to design with confidence and deploy with clarity.

We will begin by discussing the core elements of Azure governance and how they influence your ability to enforce policy, manage cost, and establish consistent telemetry across your environment. You'll understand the rationale behind dedicated subscriptions for Sentinel and how tagging, Azure Policy, and diagnostic settings play a critical role in log forwarding and segmentation.

From there, we'll move on to identity and access design. You'll learn how to implement role-based and least privilege access using Microsoft Entra ID groups, **Privileged Identity Management (PIM)**, and entitlement management. We'll explore **Unified Role-Based Access Control (URBAC)** principles and demonstrate how to structure access paths across both internal teams and delegated customer environments.

As we expand into multitenant service operations, we'll examine the delegated access models available today, such as Azure Lighthouse, **granular delegated admin privileges (GDAP)**, and B2B collaboration. Each model serves a distinct purpose, from automating cross-tenant deployments to enabling secure workbook queries and customer-facing playbooks. You'll also learn how to enforce operational controls with access packages, approvals, and access reviews to maintain governance at scale.

By the end of this chapter, you'll have a reference blueprint to architect and secure your Sentinel deployment from the ground up, while also preparing for the evolution toward unified SOC operations.

In this chapter, we'll cover the following topics:

- Designing secure and scalable Sentinel foundations using Azure subscriptions and management groups
- Building structured, least privilege access with RBAC, Entra ID, and PIM
- Deploying foundational RBAC artifacts – home/MSSP tenant
- Defining the right roles for unified control

Let's dive into designing secure and scalable Microsoft Sentinel foundations.

Designing secure and scalable Sentinel foundations using Azure subscriptions and management groups

When deploying Microsoft Sentinel, management groups are the backbone that tie governance, access, and log collection together at scale. They don't collect logs themselves, but they define how Azure Policy and diagnostic settings are inherited across subscriptions, directly shaping which signals make it into your Sentinel workspaces and which blind spots might appear. Getting this hierarchy and its policies right up front ensures that your log pipeline for Azure Policy-governed items is consistent, compliant, and ready for effective detection and response.

Management groups and log collection

When deploying Microsoft Sentinel, understanding Azure management groups is key to aligning security operations with governance and compliance. Management groups sit at the top of the Azure resource hierarchy and allow you to apply access controls, policies, and budgets across multiple subscriptions efficiently. They also play a critical role in log collection and visibility, which directly impacts Sentinel's effectiveness.

Before diving into Sentinel-specific configurations, let's start by discussing the broader Azure architecture that supports it.

Management groups and their impact on log collection

Management groups don't collect logs themselves, but they define how Azure Policy and Azure Monitor diagnostic settings are inherited. These settings are the ones that route logs to the Sentinel location. This inheritance influences where logs are collected from and, more importantly, where they're sent. If not properly configured, some logs might not reach the Log Analytics workspace connected to Microsoft Sentinel, creating blind spots in threat detection.

To avoid this, you must ensure that diagnostic settings are consistently applied across all relevant subscriptions and resources under a management group. Tools such as Azure Policy can enforce the automatic deployment of logging settings to maintain consistent log forwarding to your Sentinel-connected workspace.

For example, you may have a client that requires Sentinel instances in four regions globally due to separate SOC teams or compliance requirements. It's likely that they already enforce compliance in these four regions through Azure Policy and Microsoft Defender for Cloud, scoped to one or more management groups that encompass the Azure subscriptions for each region. Therefore, a new subscription and Sentinel instance in each region can receive logs governed by that management group's policy configuration.

Choosing the right subscription: Existing versus new

An important decision during Sentinel setup is to select where to deploy your Log Analytics workspace:

- **Existing subscriptions**: This is an ideal choice if your organization already centralizes security monitoring or has mature Azure governance. Benefits include reduced setup effort and consolidated visibility.
- **(Preferred) New secure subscription**: In greenfield scenarios or for high-sensitivity environments, it's often better to provision a new, isolated subscription purely for Microsoft Sentinel. This enhances role separation, enforces clean cost attribution, and aligns with Zero Trust principles.



When provisioning Sentinel, always tag resources consistently (e.g., Environment=Security, CostCenter=SOC) and implement RBAC early to prevent accidental exposure of sensitive data and ensure proper governance from day one.

A recommended best practice is to place all Microsoft Sentinel-related resources within a dedicated Azure subscription. If the subscription will be used for additional purposes beyond Sentinel, consider isolating Sentinel resources within a clearly defined resource group. Once provisioned, assign the appropriate Microsoft Sentinel roles, such as Sentinel Reader, at either the subscription or resource group level. This ensures consistent permission management across all resources supporting your Sentinel workspace.

Overall, management groups are foundational for structured governance, but without a clear configuration of policies and diagnostics, they can unintentionally limit Sentinel's reach. Carefully selecting a secure and scoped subscription for your deployment, either new or existing, ensures that the log pipeline is clean, secure, and segregated.



Thinking long term? Use a centralized security management group with scoped policies to onboard new subscriptions securely and consistently. If you haven't already, take a look at Azure landing zones as they provide a great foundation for scalable, well-governed cloud environments.

With your governance approach in mind, the first step is to establish the right hierarchy. That begins with creating a new management group to serve as the foundation for organizing and securing your Sentinel environment.

Creating a new management group

1. At `https://portal.azure.com/`, sign in with an account that has *Global Administrator* permissions. In the top search bar, type `Management groups` and select the **Result** button.

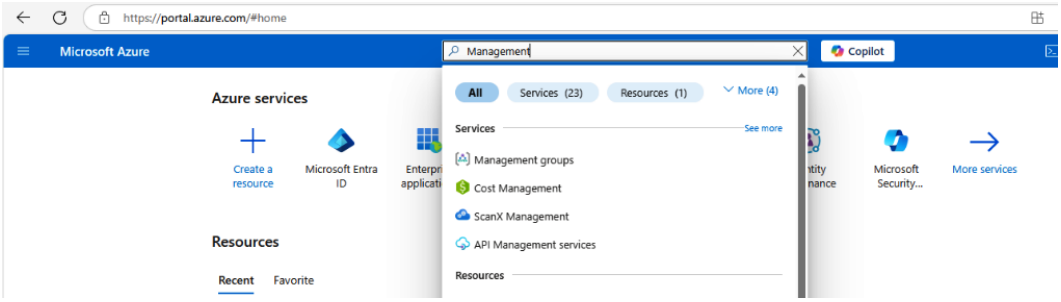


Figure 6.1: Searching for Management groups

2. Click the **+ Add Management Group** button to create a new one.

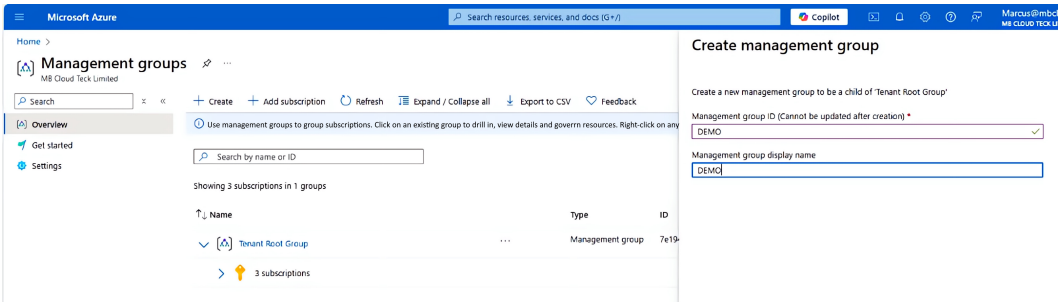


Figure 6.2: Creating a management group

3. Once created, you're ready to associate a new or existing subscription with this management group by selecting it in the drop-down list of subscriptions.

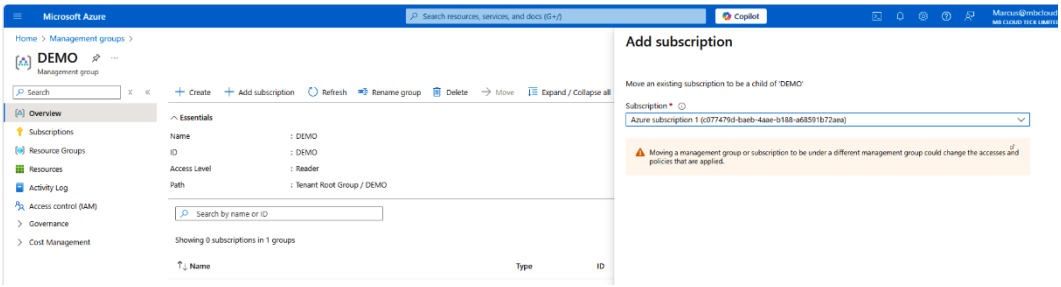


Figure 6.3: Associating the management group with the existing subscription



Management groups are essential for centrally managing access across multiple Azure subscriptions. They ensure consistent enforcement of security, compliance, and least privilege at scale. Think of it like a family tree: the management group at the top sets the baseline, and everything beneath it inherits from that structure, adding to it, not replacing it.

With your management group in place, the next step is to create a dedicated subscription for Microsoft Sentinel. This helps maintain clear boundaries, simplifies cost tracking, and ensures consistent access control across all related resources.

Creating a new subscription

To create a new subscription, you may need to contact your Cloud Service Provider (CSP) or your organization's Azure administrator, depending on how your environment is set up. If you have a direct billing relationship with Microsoft, you can typically create a new subscription using a credit card.

Follow these steps to get started:

1. Sign in to `https://portal.azure.com/`.
2. Use the top search bar to search for `Subscriptions`.
3. Select the **+ Create** button.
4. Follow the onscreen instructions to complete the setup.

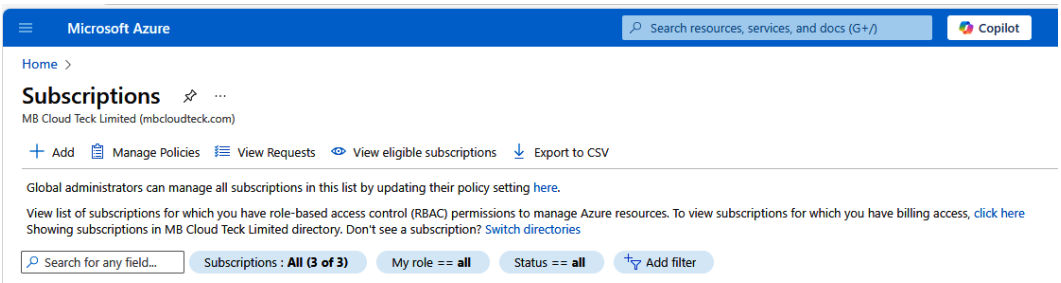


Figure 6.4: Searching for Subscriptions



Let's assume you're not just deploying this in your own environment. Even if you are, we'll base the scenario on a scalable architecture that follows best practices, especially around RBAC.

With your subscription in place, the next critical step is designing access governance.

RBAC and unified XDR access

URBAC is a centralized permission model in Microsoft Defender XDR that lets you define a single set of security roles and permissions and reuse them across multiple Defender workloads (Defender for Endpoint, Office 365, Cloud Apps, Defender for Cloud, Sentinel data lake, etc.), instead of managing separate RBAC models in each product.

In practical terms, URBAC unifies access control across Defender workloads and (in multitenant scenarios) across tenants. It simplifies role management by helping you create roles once in Defender XDR and assigning them to users and groups instead of duplicating roles per product.

URBAC also supports least privilege with granular permissions tied to SOC tasks and coexists with Entra ID roles, while still honoring global roles such as Global Admin and Security Admin.

Direct role assignments may work for small or temporary environments, but they quickly become difficult to manage and audit at scale. A mature approach would be to rely on Microsoft Entra security groups to assign permissions consistently across Microsoft Sentinel and related Azure resources.

By designing your access model up front, you enable consistent, policy-based access control, have easier onboarding and delegation through automation, and alignment with enterprise security and compliance standards. Every Microsoft Sentinel deployment must have RBAC configured to ensure that only authorized users can access and manage sensitive data. Currently, access to both Sentinel and the connected Log Analytics workspace is managed in the Azure portal using Entra security groups.

RBAC assignments can be scoped at the subscription, resource group, or resource level, offering fine-grained control. In specific use cases, table-level access can also be configured. *Figure 6.5* shows how access for a specific team to specific tables could be granted.

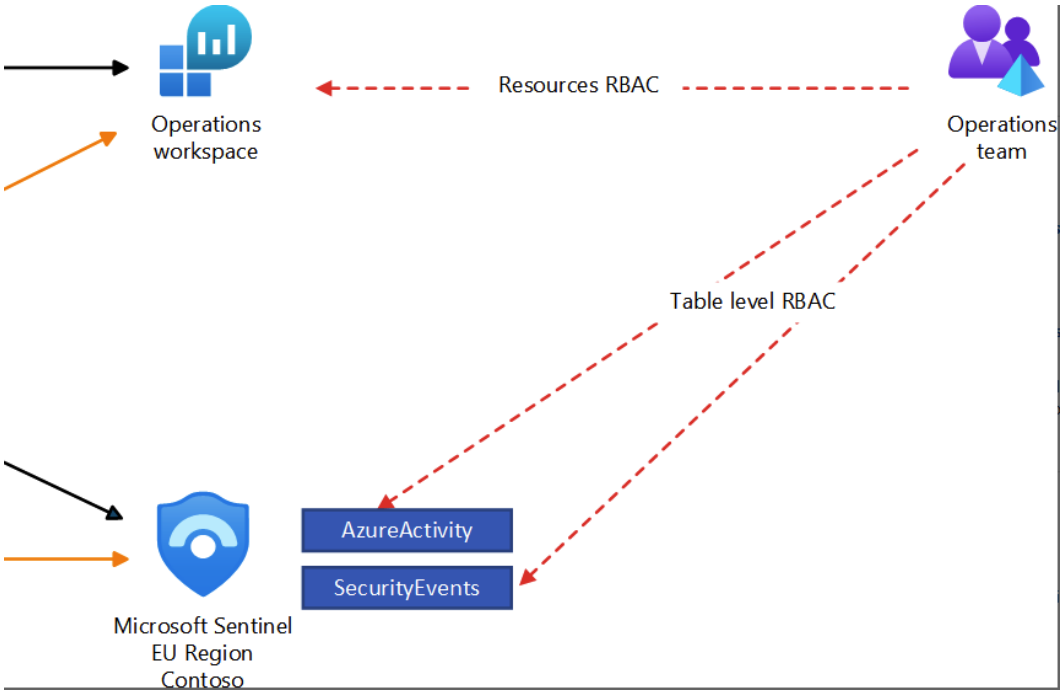


Figure 6.5: RBAC assignments showing access direction

Microsoft is moving toward a unified security operations platform, and RBAC for Sentinel is expected to be integrated into the Microsoft Defender XDR experience. This will bring role alignment across Defender, Purview, and other security services under a common access model. To prepare for this transition, align your current Entra roles to operational SOC functions such as *Tier 1 Analyst*, *Threat Hunter*, and *Security Administrator*. This groundwork will ease the adoption of the URBAC model as it becomes more widely available.



Even with the recent introduction of **Microsoft Tenant Organizations (MTO)**, most real-world, multitenant scenarios, particularly for MSSPs, still depend on Azure Lighthouse for service principal delegation, cross-tenant automation, and workbook access.

While Microsoft is moving toward a more unified access model, Azure Lighthouse and B2B collaboration remain critical for many current operational scenarios, especially in multitenant environments. These models provide the flexibility and control needed for MSSPs and large enterprises to securely manage resources across tenant boundaries.

Key scenarios where Azure Lighthouse or B2B remains essential

Despite the emergence of newer multitenant management options such as MTO, there are still several critical use cases where Azure Lighthouse or B2B collaboration is required to maintain operational effectiveness:

- **Cross-tenant service principal delegation**: Azure Lighthouse enables MSSPs to delegate permissions to service principals, which is essential for automating tasks such as content deployment via CI/CD pipelines in Microsoft Sentinel. This capability is not supported by MTO, as it relies on B2B invitations, which do not extend to service principal identities.
- **Cross-tenant workbook queries**: Azure workbooks use the Log Analytics API and the `workspace()` function to pull data from multiple workspaces. When those workspaces exist across different tenants, Azure Lighthouse is required to enable the necessary cross-tenant data access for seamless workbook functionality.
- **Automation and playbooks**: Advanced automation scenarios, such as running playbooks across tenants, still depend on Azure Lighthouse. For example, an MSSP may want to host playbooks in its own tenant to protect intellectual property while executing them within a customer's environment. Azure Lighthouse uniquely supports this kind of secure, delegated execution.

Role	View and run playbooks	Create and edit playbooks	Create and edit analytics rules, workbooks, and other Microsoft Sentinel resources	Manage incidents (dismiss, assign, etc.)	View data, incidents, workbooks, and other Microsoft Sentinel resources
Microsoft Sentinel Reader	----	----	----	----	✓
Microsoft Sentinel Responder	----	----	----	✓	✓
Microsoft Sentinel Contributor	----	----	✓	✓	✓
Microsoft Sentinel Playbook Operator	✓	----	----	----	----
Logic App Contributor	✓	✓	----	----	----

Figure 6.6: Table with Sentinel roles

Having explored the essential cross-tenant access scenarios supported by Azure Lighthouse and B2B, it's time to focus on how access is managed within your environment, whether for internal teams or managed client tenants.

Building structured, least privilege access with RBAC, Entra ID, and PIM

In this section, we will discuss how you can achieve a complex but formalized access approach in practice. Let's start by establishing foundational access controls.

Establishing foundational access controls

One of the first steps in building a mature and auditable Microsoft Sentinel and unified security operations environment is to define a consistent approach to access management, starting with foundational security groups.

We'll walk through a practical example using a well-structured RBAC model that incorporates security groups, access packages via entitlement management, access reviews, and Entra ID PIM. This framework ensures that access to sensitive security tools and roles is governed by the principle of least privilege, while also satisfying enterprise requirements for approvals and audit trails.

You might be asking, "*Why create a separate access package and PIM group for each team, especially when some roles seem to overlap?*"

In a single tenant, single client setup, a simplified approach often works well. A single access package might include multiple role-based groups, and users can elevate into the roles they need through PIM, all managed in one centralized, controlled environment.

However, things change significantly in a multiclient model, particularly when working under Azure Lighthouse or GDAP. In these scenarios, each client environment must be treated as a separate security boundary, with its own set of delegated access controls. Security groups are typically created in your home tenant to manage these delegations. While a user may have permission to elevate into a role such as Sentinel Contributor (referred to here as *Role 2*), they should never be able to activate that role across multiple clients simultaneously. This is because of the following reasons:

- **Access scope separation**: Each client represents a distinct trust boundary. Cross-client access violates this separation and increases risk.
- **Audit and compliance**: Client-specific access packages let you track who requested what, when, and for which customer. It is essential for regulated environments.
- **Granular control**: With individual access packages and PIM groups, you can enforce tailored approval workflows and role activation policies per client.
- **Least privilege enforcement**: Users only request and activate roles relevant to the specific client they're supporting. Nothing more, nothing less.

While unified access models may work in smaller or isolated environments, multiclient operations demand precision and separation. This layered approach, using per team and per client access packages and PIM groups, supports a secure, scalable, and fully auditable delegation framework.

This brings us to the core principle behind URBAC in Microsoft Sentinel and the broader Microsoft security ecosystem: least privilege by design.

Overall URBAC concept of least privilege

Even if a user is a member of a designated security team, they won't automatically receive privileged access. Instead, access must be explicitly requested through an access package defined per team. Once approved, the user must still elevate into a specific role using PIM to perform sensitive tasks.

This layered model ensures a clear separation between the following:

- Group membership (who the user is aligned with)
- Access entitlement (what roles they're eligible for)
- Privilege elevation (when and how they can activate access)

It provides full traceability and enforces tighter security boundaries, critical in both regulated environments and Zero Trust architecture. You can also tailor access packages to support separation of duties and insider risk management, ensuring that users only get the access they need when they need it.

Figure 6.7 shows an example of this model in practice within an MSSP scenario, using Azure Lighthouse connections to client tenants and tightly scoped group-to-role mappings, ensuring that access is limited to what's needed in each delegated environment.

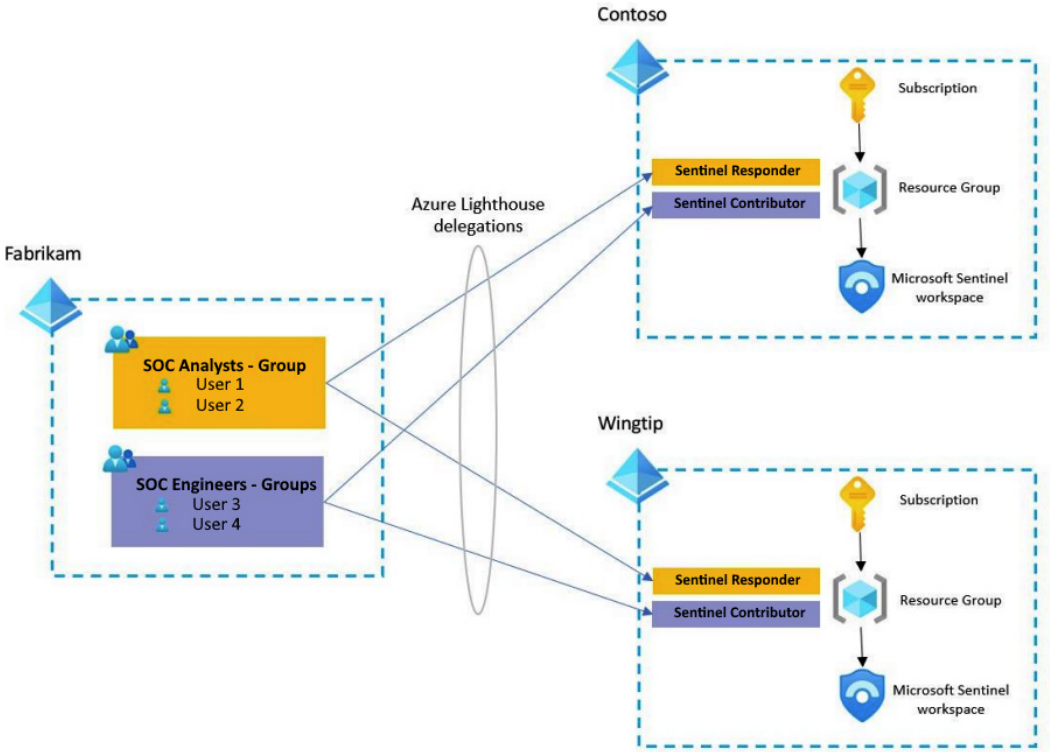


Figure 6.7: MSSP scenario with Lighthouse connection

Now that we've covered the core concept of least privilege and how URBAC enforces layered access, let's bring it to life with a practical example.

Scenario: Team-based role design for Sentinel operations

To illustrate how this model works in practice, we'll use a hypothetical organization with the following security-focused teams:

- **Security audit**: Internal auditors
- **Security responders**: SOC analysts
- **Automation team**: SOAR engineers and playbook developers
- **Security engineers**: Cloud infrastructure security team

We'll also assume that this organization has both internal access needs and responsibilities to manage partner or client environments using Azure Lighthouse and GDAP. In this scenario, the security team must not only access their own internal resources but also securely manage one or more client environments (or "subcompanies").

This setup requires an access model that supports delegated access to external tenants, while also maintaining strict separation of duties, least privilege, and clear auditability across both internal and client environments.

For clarity in this section, we'll use the following terms:

- **Home tenant**: The organization's internal tenant
- **Client tenant**: The customer or partner tenant being managed

Each security team should have a dedicated Entra ID security group that reflects its function. These groups will be mapped to specific roles and resources based on operational responsibilities. If your organization hasn't defined these groups yet, now is the time. Start by looking at existing mail-enabled or Microsoft Teams groups used for departments, as these can often be repurposed.

We'll also assume there's an overarching cloud security team group that governs who is eligible to request access via access packages. This group could be a Microsoft 365 group used for broader communication and collaboration across security teams. This restricts the rest of the company's ability to see the following access packages.

Finally, you'll need a group for **approvers,** typically composed of managers or team leads. We'll refer to this group as **SG SecurityApprovers**. With the access model defined and team roles established, the next step is to deploy the foundational RBAC artifacts that bring this model to life, starting with the creation of security groups in your home tenant.

Deploying foundational RBAC artifacts – home/MSSP tenant

We will now create the basic foundations and collection of artifacts that will help create granular mappings. In your own scenario (separate from the one we are giving as an example), this will be tailored to your potentially extensive requirements based on least privilege and organizational complexity.

Creating security groups

To create security groups, begin in your **home tenant**, where you can find the central access logic:

1. Navigate to the **Microsoft Entra admin center**: `https://entra.microsoft.com`
2. Select **Groups** from the left-hand menu, then click **+ New group**.

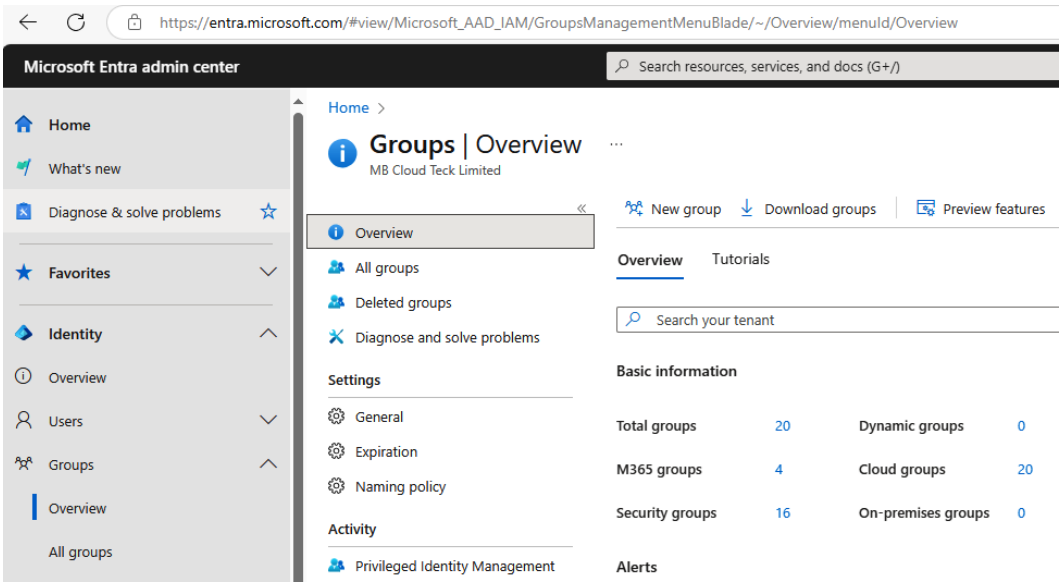


Figure 6.8: Creating a new security group

3. Create one security group per role-based team. Use the following configuration:
    - **Group type: Security**
    - **Microsoft Entra roles can be assigned to the group: No**
    - **Membership type: Assigned** (or **Dynamic**, if appropriate for your environment)

Here are some example group names:

- `SG SecurityAudit`
- `SG SecurityResponders`
- `SG AutomationTeam`
- `SG SecurityEngineers`

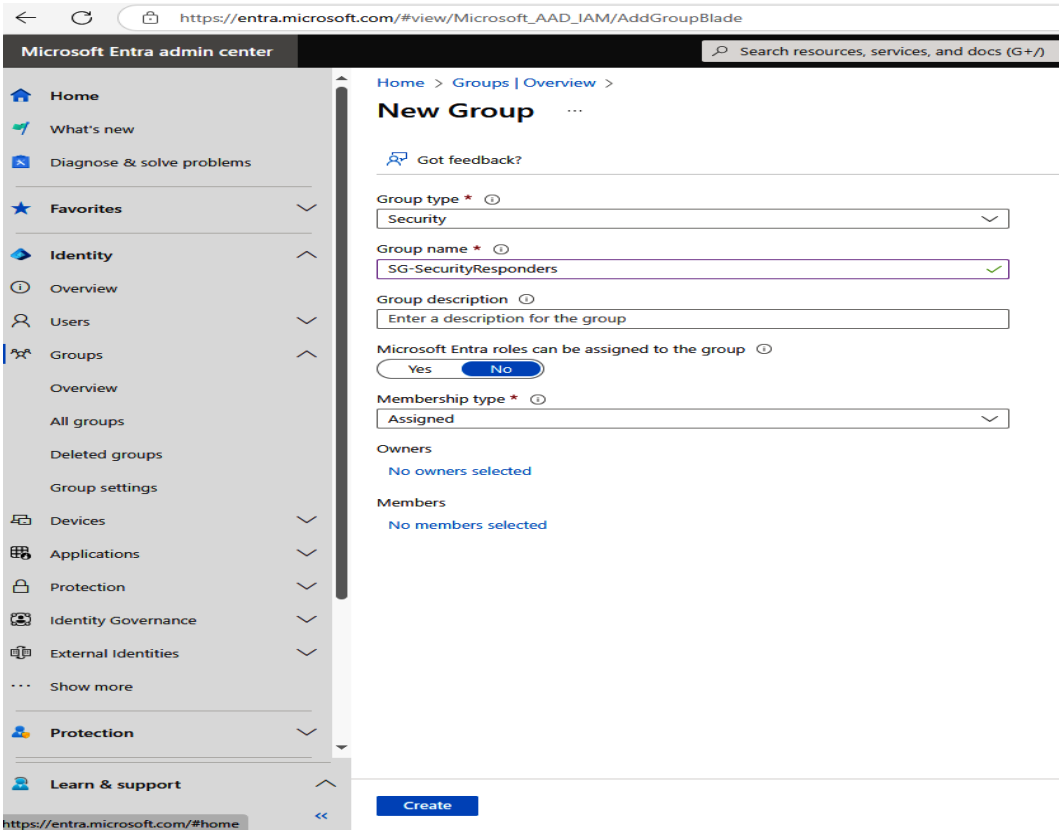


Figure 6.9: Group settings



You can create additional groups as needed to allow for more granular control, for instance, to separate Microsoft 365 roles from Azure-specific ones. For this demonstration, we'll keep things focused on the core teams.

4. Once completed, you should now have four new security groups ready to be mapped to access packages and PIM assignments.

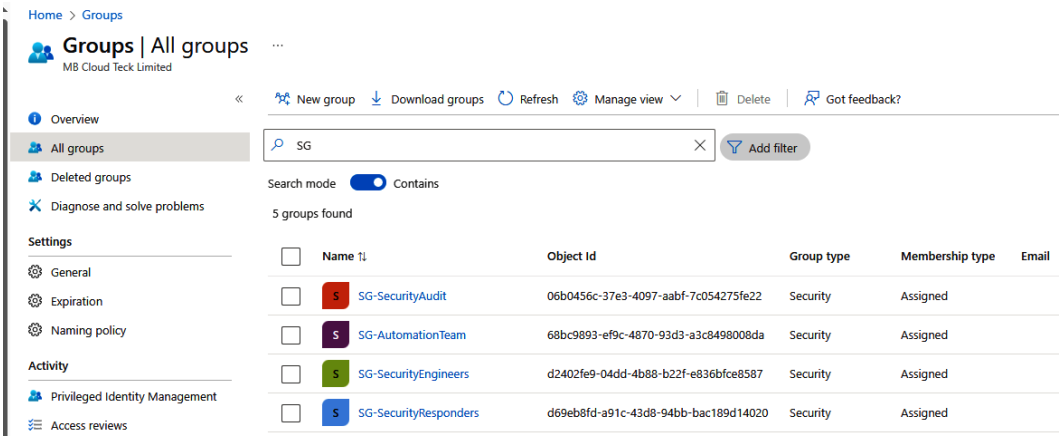


Figure 6.10: Security groups created for mapping

Now that the security groups are created, let's move on to creating an access package.

Creating an access package

An access package is required to allow a group of users to request access to a resource. In this case, the resource will be the security group.

Follow these steps to create a new access package:

1. In Entra, navigate to **Identity Governance | Entitlement Management | Access packages | New access package**.

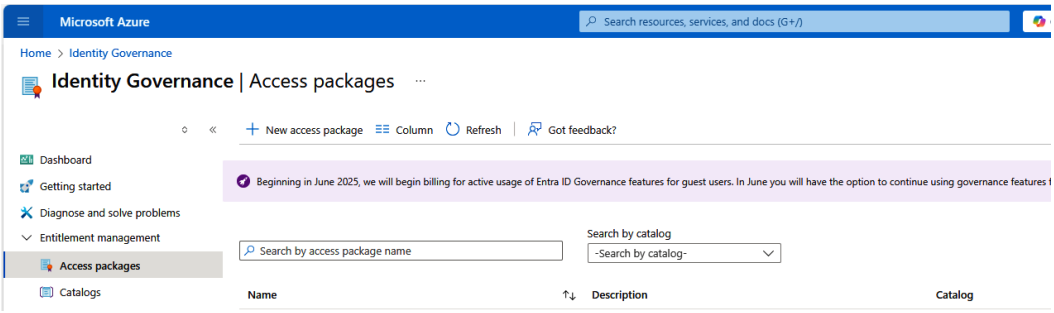


Figure 6.11: Creating a new package

2. Create one access package per team, such as the following:

    - `AP SecurityResponders`
    - `AP SecurityAudit`
    - `AP AutomationTeam`
    - `AP SecurityEngineers`

    You can create multiple access packages depending on the resources, security groups, and roles they will grant access to. For this example, we'll keep it simple with four core packages.

3. Name and describe each package clearly so users and approvers understand its purpose.

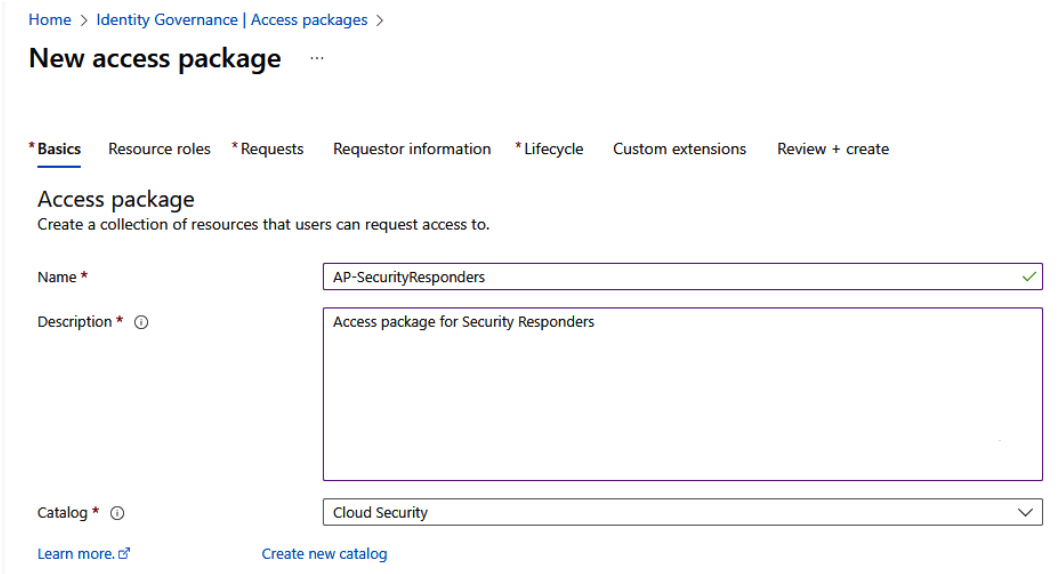


Figure 6.12: New package configuration

4. Once that is done, navigate to the **Resource roles** tab. Add the corresponding security group (e.g., `SG SecurityResponders`) and set the role as **Member**.

These are the groups that the requesting user will be made a part of when approved into the access package.

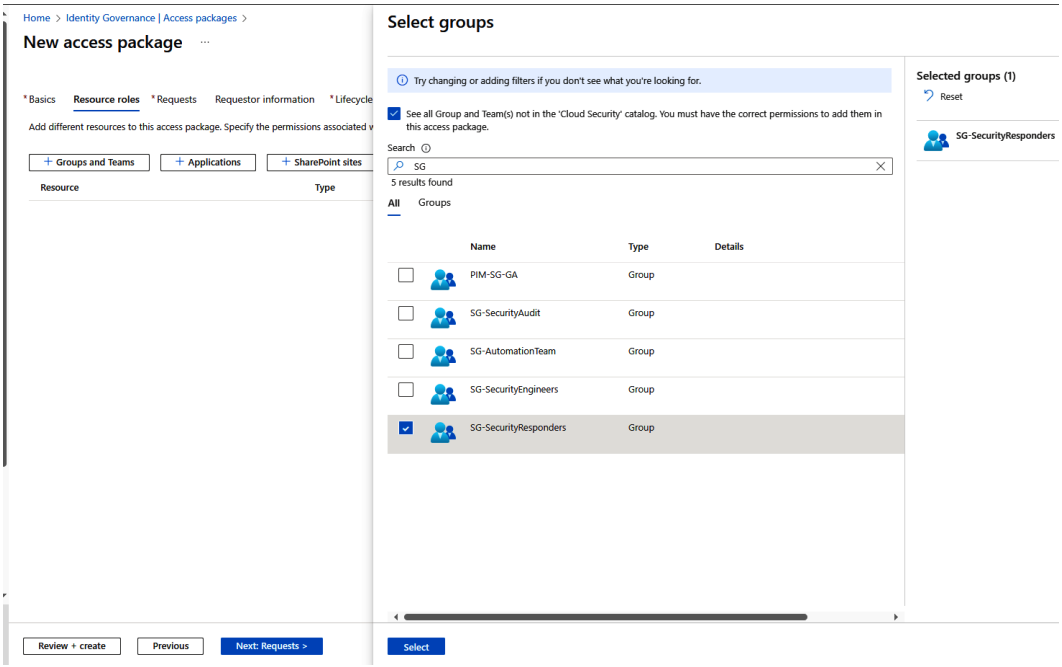


Figure 6.13: Available groups for the requester once approved

5. Update the request settings as follows:

    - Allow users in your directory to request access
    - Restrict requests to specific users and groups (e.g., your overarching cloud security team)
    - Configure an approval workflow, such as routing to a manager or designated security owner

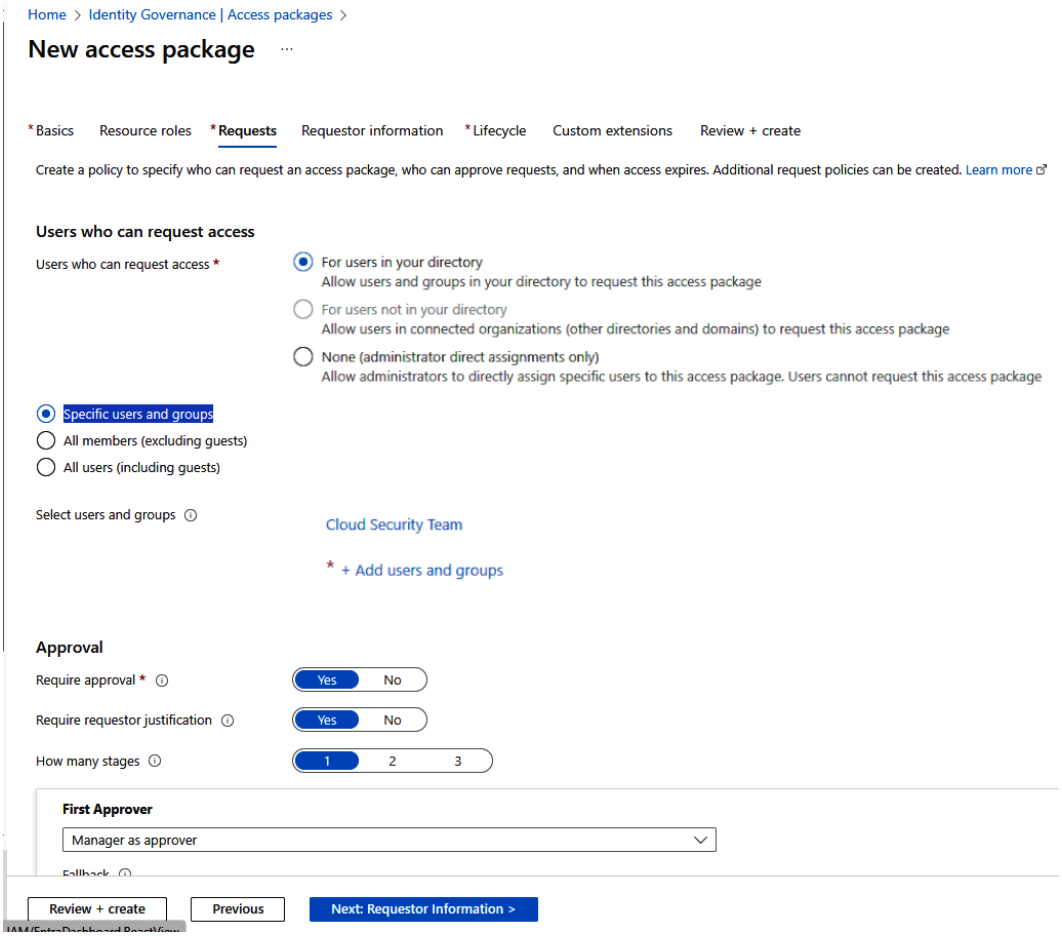


Figure 6.14: Access package Requests settings

6. Require justification for access and enable notifications to keep requesters and approvers informed.

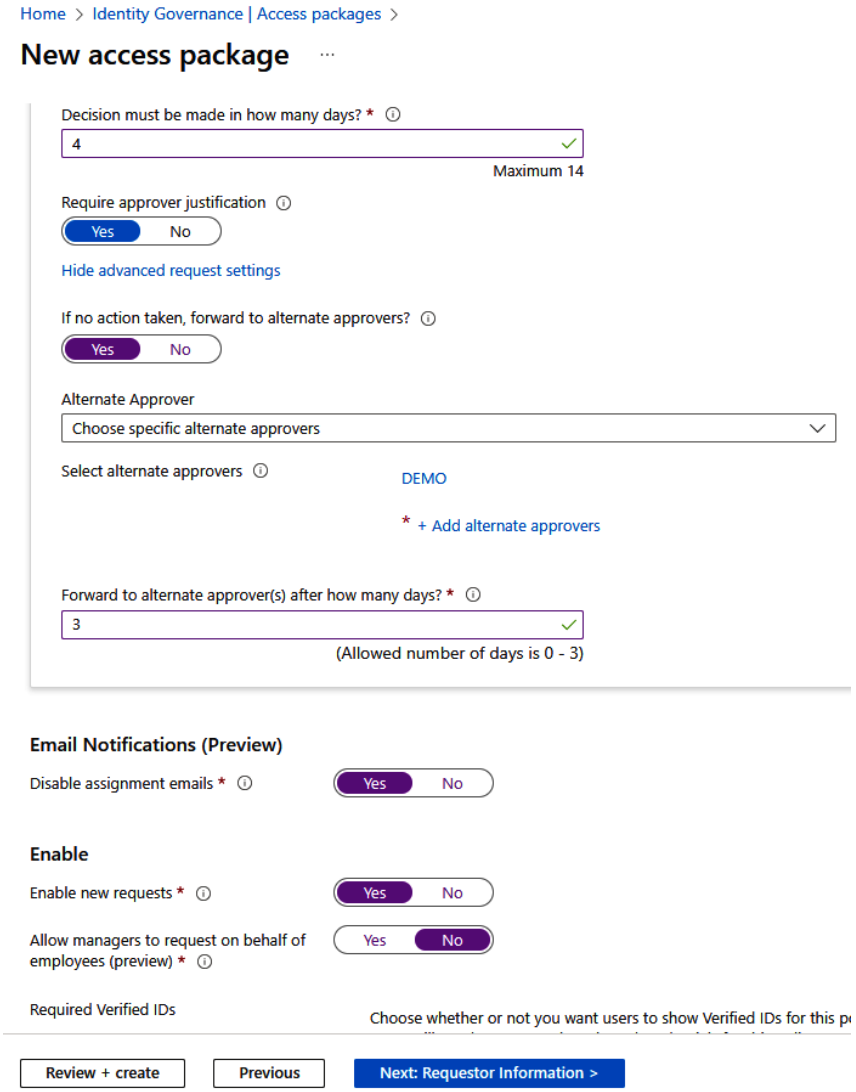


Figure 6.15: Access packages recommended settings

With the core settings defined, the final step in configuring your access package is to manage its lifecycle settings. These settings control how long access is granted, how it's reviewed, and how approvals are maintained over time.

All lifecycle settings can, and should, be tailored to meet your organization's specific security, compliance, and operational requirements.



The overhead of managing access packages, especially those with frequent validations or multilevel approvals, can be high. We recommend starting small, perhaps with a single approver and quarterly access reviews, then scaling up as your processes mature. Keep in mind that access packages also introduce a cultural shift. Bring users along gradually to ensure the adoption of this more secure and auditable access model.

You can create as many access packages as needed, each with its own configuration. For example, one package might provide time-limited access to an admin tool with multiple approval stages and weekly access reviews, while another could be more lightweight for short-term project access.

7. Update the lifecycle settings as per your requirements. The following are a few suggested lifecycle settings:

    - **Access duration**: Set a default access period (e.g., 90 days)
    - **Access reviews**: Enable periodic reviews to ensure that entitlements remain valid and appropriate

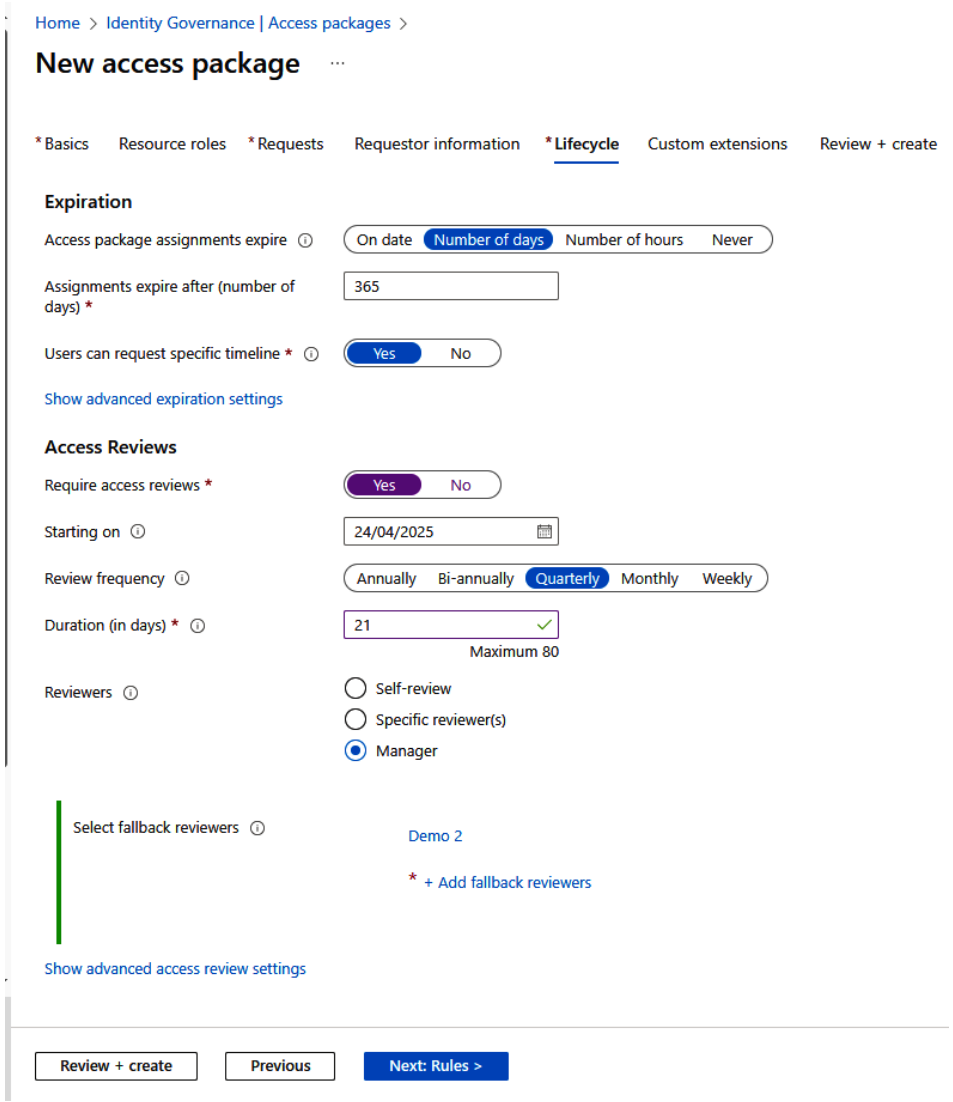


Figure 6.16: Access extended recommended settings

8. Once your configuration is complete, publish the access package to make it available for eligible users to request.

Now that we have some access packages provisioned, let's create some groups to assign to these.

Defining role-based access groups

Now, we need to define the role-based access client groups, which we'll manage using PIM at different levels. We'll define two sets: one internal only (home tenant) and one for external scenarios (client tenant), as follows.

For the internal scenario, ensure that Microsoft Entra roles **can** be assigned to these groups. Note that once set, this cannot be changed.

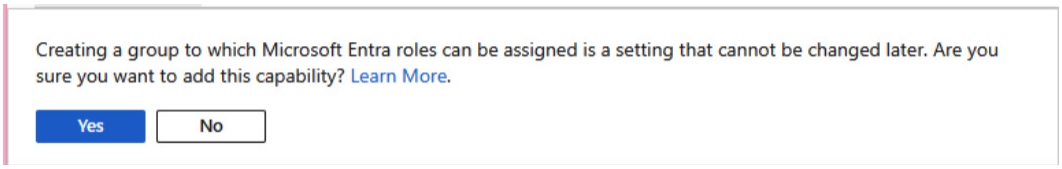


Figure 6.17: Access packages extended settings

Here are a few roles that can be added to the group in this scenario:

- `HomeTenant_Tier1_Reader`
- `HomeTenant_Tier2_Responder`
- `HomeTenant_Tier3_Engineer`
- `HomeTenant_Tier4_Automation`

For an external-based client tenant scenario, let's look at a typical scenario that an MSSP or a large enterprise with multiple tenants will face when onboarding an external tenant into GDAP or Lighthouse.

Do **not** enable the Entra roles feature in this scenario. No direct roles are required; therefore, this feature is not needed, and if enabled, poses a security risk. Access should be granted using one of the external methods described later in *Chapter 7*.

Here are a few roles that can be added to the group in this scenario:

- `ClientTenantContoso_Tier1_Reader`
- `ClientTenantContoso_Tier2_Responder`
- `ClientTenantContoso_Tier3_Engineer`
- `ClientTenantContoso_Tier4_Automation`

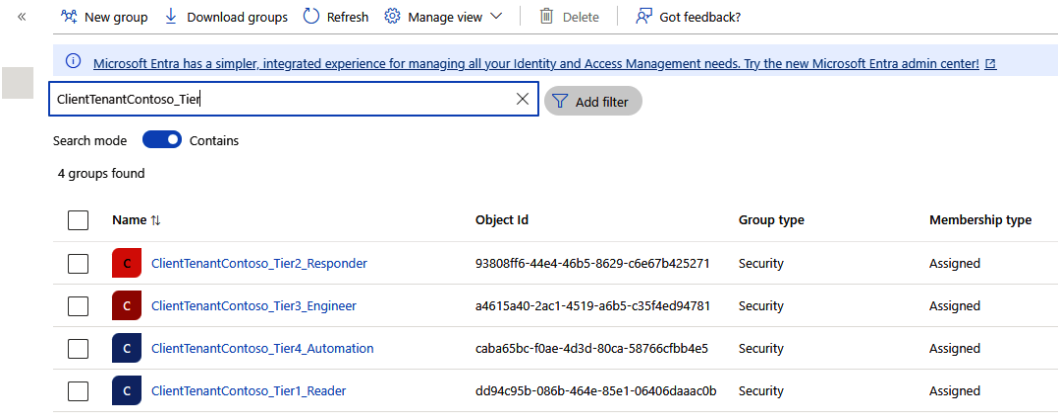


Figure 6.18: Access packages external scenario

Enabling PIM

You have to configure PIM for each home or client access group. This is done to ensure that elevation requests follow an approval workflow. The following steps demonstrate this process for one group:

1. Navigate to **Privileged Identity Management** under the **Activity** section on the left-hand menu.
2. Select the desired access group, then enable PIM.

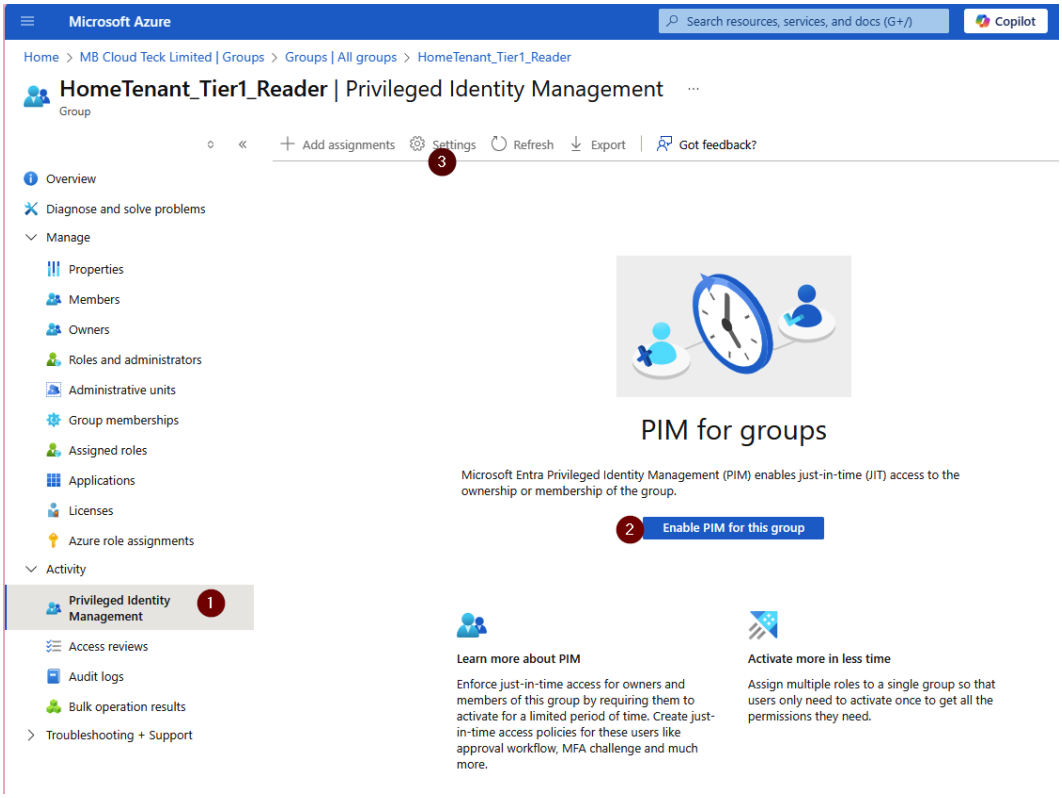


Figure 6.19: Configuring PIM for each access group

3. Allow the activation process to finish before moving on. You can repeat these steps for each additional access group as necessary.

Next, we need to allow permanent eligible assignment in the **Member** settings. This ensures that users included in the access package will always have the option to activate the role, without needing to be re-added periodically.

Follow these steps to configure this:

1. Click the settings (cog icon) at the top of the page and select **Member settings**.
2. Enable the **Allow permanent eligible assignment** option.

3. Also, configure the following settings:

    - Require Azure MFA on active assignment.
    - Require approval on activation. Select the approvers group we specified earlier.
    - Allow permanent eligible assignment.
    - Expire active assignments after 15 days.
    - Require justification on active assignment.

    Your PIM member settings should resemble the following configuration:

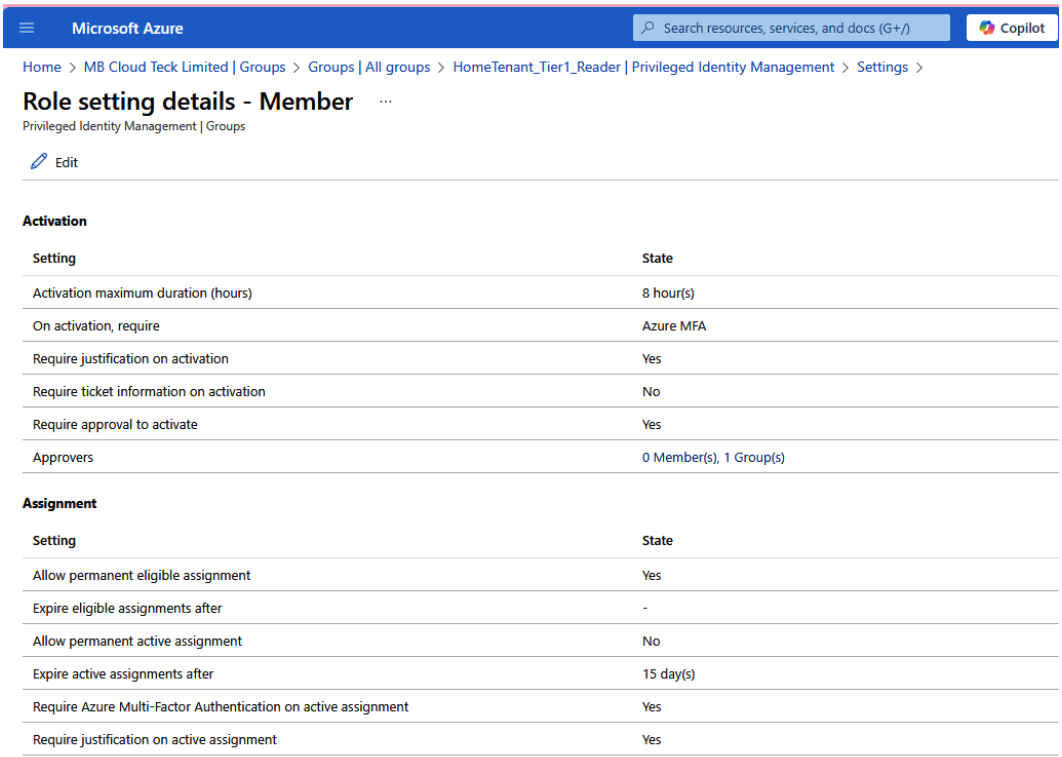


Figure 6.20: PIM recommended settings

We'll now grant our previously defined role-based security groups, along with any future members' visibility and access to request the eligible PIM groups.

1. Navigate to your first home or client group.

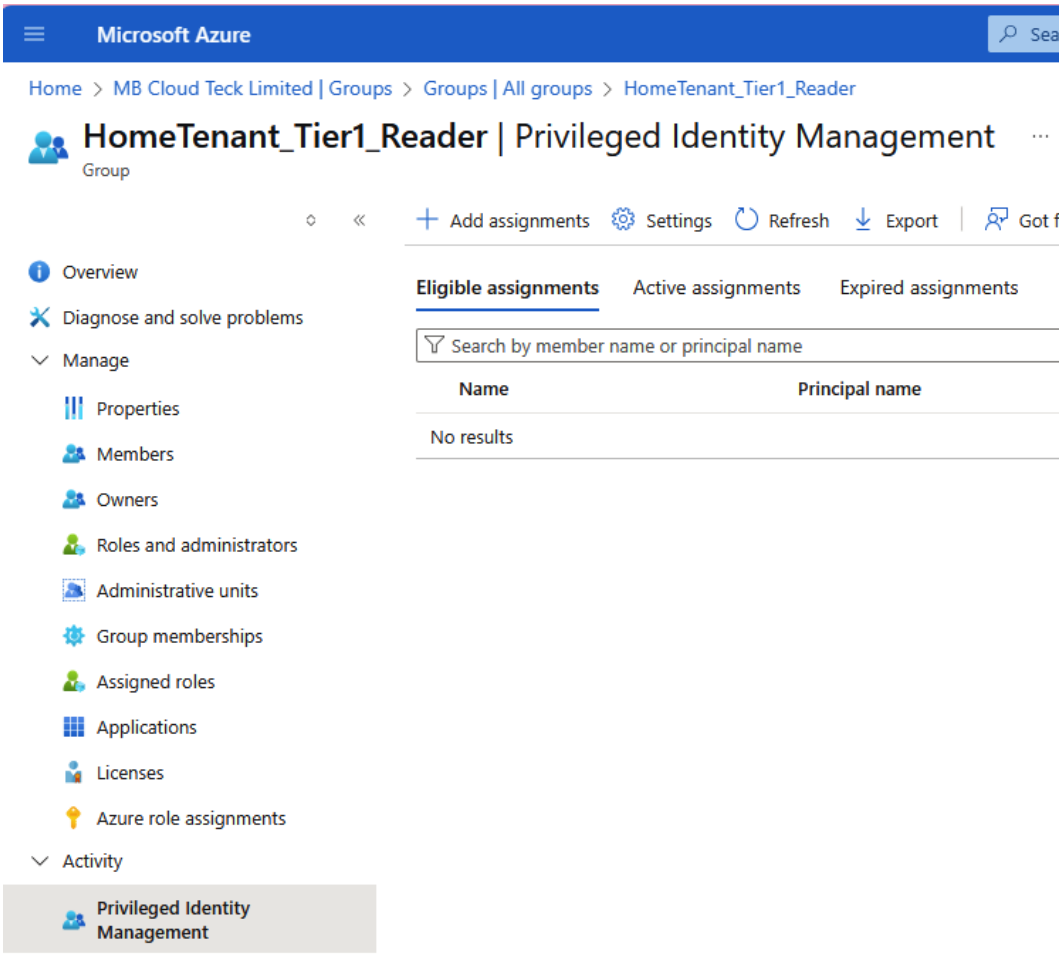


Figure 6.21: PIM security group settings

2. Click the **+ Add assignments** button and add an eligible assignment. Search for the relevant security group (in this example, `SG SecurityAudit`).
3. Set the assignment to **Permanently eligible**, then click **Assign**.

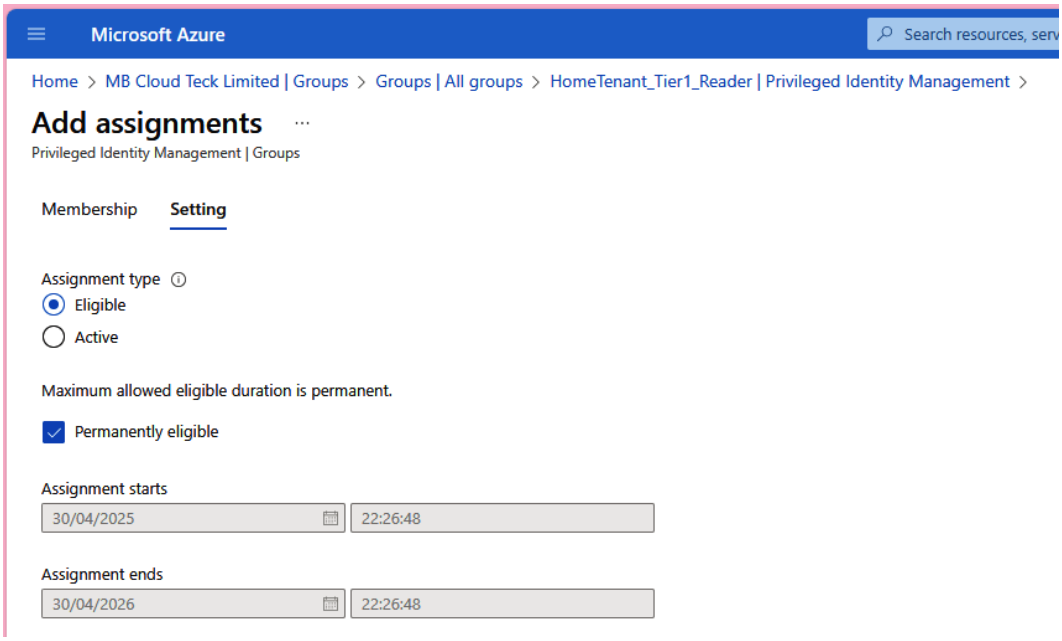


Figure 6.22: PIM adding assignment type and duration

4. Repeat this process for all required role-based security groups and their corresponding eligible PIM groups.

Once you have replicated this for all the potential roles, you will be ready to add some M365 global roles.

Table 6.1 shows the traceability of how the access package roles and approvals will translate.

Access package	Role-based security group	Eligible PIM group
`AP SecurityAudit`	`SG SecurityAudit`	`HomeTenant_Tier1_Reader` `ClientTenantContoso_Tier1_Reader`
`AP SecurityResponders`	`SG SecurityResponders`	`HomeTenant_Tier2_Responder` `ClientTenantContoso_Tier2_Responder`
`AP AutomationTeam`	`SG AutomationTeam`	`HomeTenant_Tier3_Engineer` `ClientTenantContoso_Tier3_Engineer`
`AP SecurityEngineers`	`SG SecurityEngineers`	`HomeTenant_Tier4_Automation` `ClientTenantContoso_Tier4_Automation`

Table 6.1: Access package to role with PIM group requirement example

Let's review what we have done until now. We have created access packages to manage approvals for specific roles. Then, we set up client-specific security groups (for both home and client tenants) and enabled them for PIM. These groups were assigned as permanently eligible within their respective roles.

From the user's perspective, the experience looks like this:

1. They request access via an access package.
2. This places them into the appropriate security group based on their role.
3. As members of these groups, they gain visibility of PIM-enabled client groups, allowing them to select the client environment they need and activate permissions according to the PIM configuration.



Looking ahead

Microsoft is working on a new method to directly assign roles through access packages. However, as of now, this feature is still in private preview and lacks the granularity required for scenarios involving multiple client mappings, Microsoft 365 and Azure role assignments, and delegated access via GDAP and Lighthouse. This limitation stems from the fact that security group object IDs are tightly linked to GDAP requests and Lighthouse delegations.

Follow this link for more details: `https://learn.microsoft.com/enus/entra/idgovernance/entitlementmanagementroles`.

While features such as role assignable groups in Microsoft Entra ID can technically be used for role assignment, they often reduce visibility. Instead of assigning roles directly to users, you're managing group membership, making it harder to track who has what access.

By contrast, entitlement management provides a more transparent approach by assigning Microsoft Entra roles to both employees and external users and easily reviewing user entitlements to see assigned roles. It also defines roles as eligible or active within an access package for greater control and auditability.

What about roles?

At this stage, simply being a member of the security groups we created does not grant any permission. This is where we circle back to the two sets of groups we defined earlier: one for your home tenant (with Entra role assignable enabled at creation) and one for your client tenants.

For the home tenant, you'll now begin assigning Microsoft 365 and Azure roles directly to the corresponding security groups. For Entra ID global Microsoft 365 roles, these can be assigned directly to the role-enabled groups in your home tenant.



In the future, Defender-specific URBAC roles will be managed directly within the unified security operations platform.

Let's walk through two examples of role assignment for the home tenant.

Option 1: Assigning roles in the unified security operations platform

Within the unified platform, you can import or create custom roles that provide the same effective access level as traditional RBAC roles. This approach is ideal for Defender products and is recommended because it supports granular, role-based control across services.



As Defender roles continue to evolve, check the official documentation when assigning roles to ensure alignment with the latest capabilities: `https://learn.microsoft.com/en-us/defender-xdr/compare-rbac-roles#microsoft-entra-global-roles-access`.

First, we will navigate to the `HomeTenant` tier 1 group as an example, and check for the assigned roles. We will then add the global roles that translate to the Defender XDR URBAC platform.

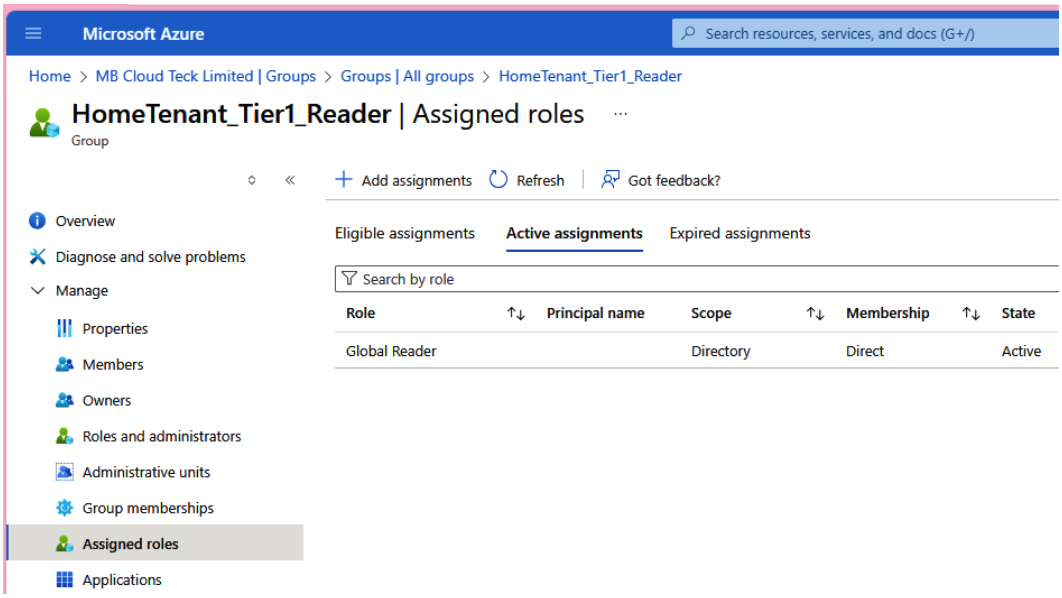


Figure 6.23: Assigning roles in unified SecOps

As you can see in *Figure 6.24*, the custom role can then be created with the assigned mapped group.

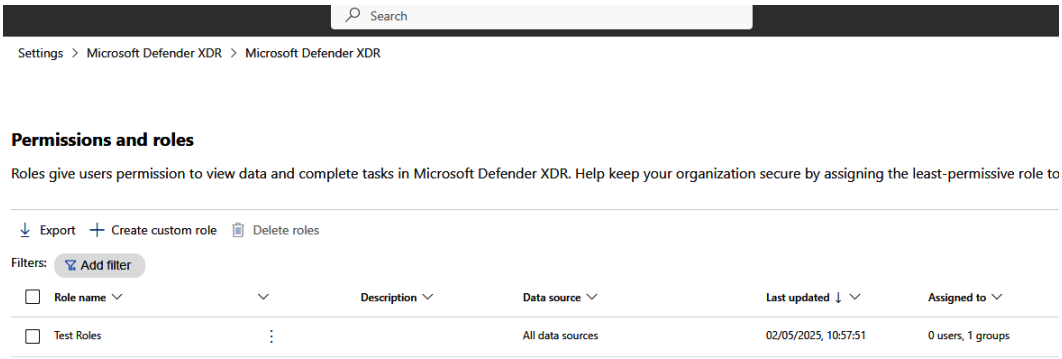


Figure 6.24: Roles in Defender XDR

You can then review the permissions and assignments and click **Submit** to add the assignments.

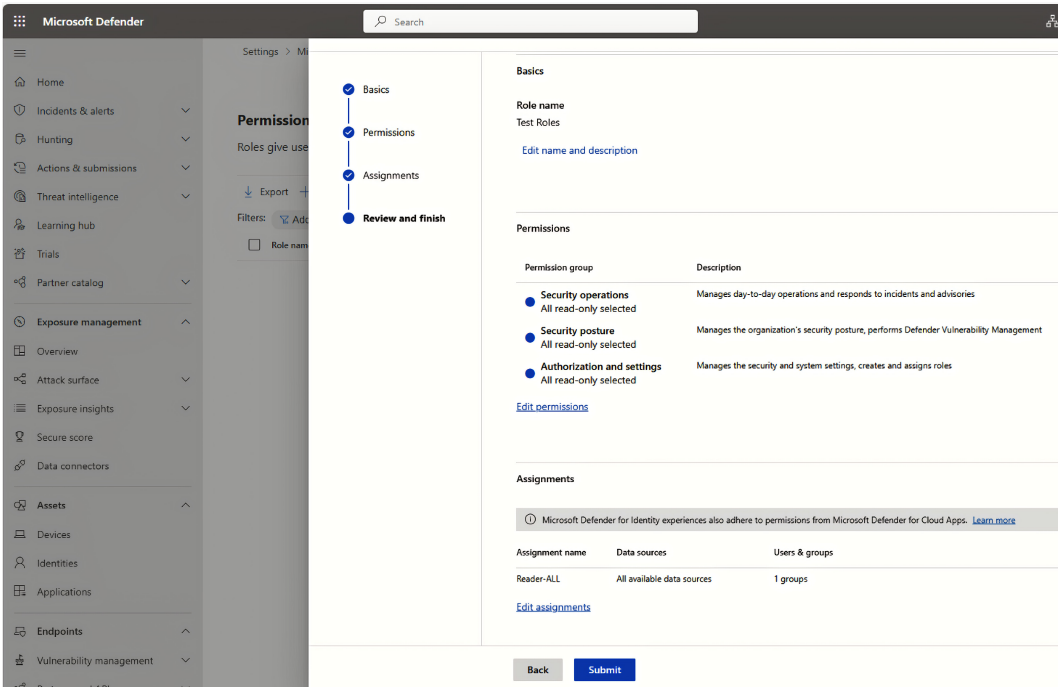


Figure 6.25: Adding assignments in Defender XDR

This simple way to map the roles provides a constant granular approach across the Defender XDR URBAC platform. Next, let's look at the Azure side of the platform.

Option 2: Assigning Azure roles via IAM

For Azure roles, navigate to the IAM settings of the relevant subscription or resource group (ideally, the security landing zone created for Microsoft Sentinel), and assign the appropriate roles to your groups. Click **Submit**.

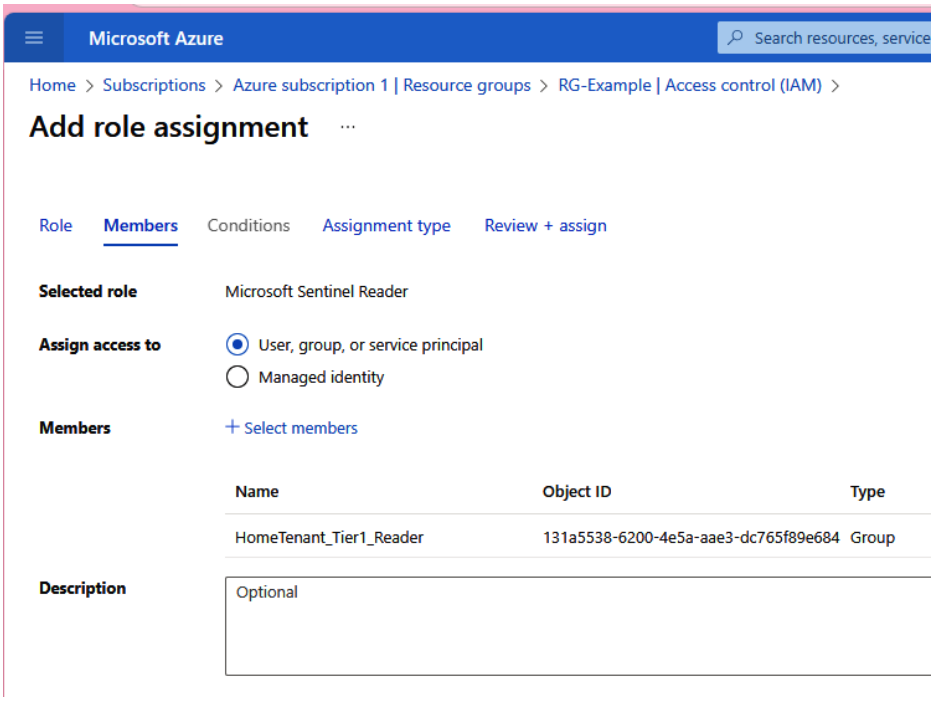


Figure 6.26: Assigning roles via IAM

You've now completed the setup of a scalable and granular RBAC mechanism for your home tenant. We haven't covered sub-clients yet. Before we move on, let's review some of the recommended roles you may want to assign to your security groups, either directly or using methods we'll explore next.

At this point, you're in the right groups, but you still need to decide how to assign the roles directly, via GDAP, using Lighthouse, or through B2B collaboration. Let's break down each option in the following section.

Defining the right roles for unified control

Let's understand the importance of URBAC through an example access design, which follows best practice principles such as least privilege and explicitly verifies the granular groups.

Setting up RBAC/Azure RBAC

Now that you've set up your RBAC framework, it's time to define the actual roles that will govern access across your environment. So far, you've created the necessary security groups and configured PIM, but group membership alone does not grant permissions. To enable functionality, we need to assign Microsoft 365 and Azure roles to these groups, starting with your home tenant.

In this section, we'll walk through a basic home tenant scenario, assigning roles directly to the groups we created earlier. We'll explore the more complex MSSP/client tenant use cases shortly.

The roles listed here are illustrative, based on real-world deployments. They're designed to cover both operational and Business As Usual (BAU) access needs, such as those for SOC analysts, responders, engineers, and automation specialists. Think of this as a starting point for building a flexible, scalable role matrix.

Example role matrix for home tenant groups

For example, your home tenant groups can be assigned to these roles:

Group name	**Microsoft 365 roles**	**Azure roles**	**Persona**	**Why?**
`ClientX_Tier1_Reader`	Global Reader	Microsoft Sentinel Reader + Reader	Auditor	View Sentinel incidents and read all resources in the delegated tenant
`ClientX_Tier2_Responder`	Security Operator	Microsoft Sentinel Responder	SOC Analyst	Investigate and respond to incidents
`ClientX_Tier3_Engineer`	Security Administrator	Microsoft Sentinel Contributor + Contributor	Security Engineer	Manage Sentinel and broader Azure resources
`ClientX_Tier4_Automation`	Security Operator	Logic Apps Contributor, Microsoft Sentinel Contributor + Contributor	Automation Engineer	Modify automation rules and manage Logic Apps in separate resource groups

Table 6.2: Home tenant role matrix

Roles will evolve as teams grow, tools change, and responsibilities shift. That's why we've spent time building a modular and scalable RBAC structure. You can easily swap roles in and out, create new groups, or define more granular PIM configurations as your environment matures.



Azure roles are still required for managing Microsoft Sentinel, even as the URBAC model continues to expand. Stay up to date with the latest Microsoft Defender RBAC guidance:

`https://learn.microsoft.com/enus/defenderxdr/comparerbacroles#microsoftentraglobalrolesaccess`.

Common role-based requirements

When using Sentinel, the most common roles required are included in *Table 6.3*. These should be understood when assigning roles to the team, with the least privilege principle in mind.

Task/Requirement	Required roles and permissions	Why this is needed
Connect data sources to Sentinel	Write permissions on Sentinel workspace; connector-specific permissions	Configure connectors, some of which need additional API or service access
Manage out-of-the-box content	Sentinel Contributor (resource group level)	Content hub deployments require Contributor access for templates and resources
Automate responses (playbooks)	Run: Sentinel Playbook Operator Create/Edit: Logic App Contributor	Running is lightweight; editing requires access to Logic Apps, which are separate Azure resources
Allow Sentinel to run playbooks	Grant service account Owner or Contributor rights on playbook resource group	Enables system identity to execute automation securely and without user involvement
Allow guest users to assign incidents	Sentinel Responder + Entra Directory Reader	Guest users need Directory Reader to view user profiles and assign incidents
General Azure and Sentinel infrastructure setup	Azure: Owner/Contributor/ ReaderLog Analytics: Contributor/ Reader	Required for initial setup and ongoing access to core Azure and Sentinel infrastructure

Table 6.3: Common required Sentinel roles

Moving forward: Multitenant role assignment and delegated access

With your Entra ID tenant, RBAC, and group structure in place, and with a clear understanding of role mappings, the next step is to extend secure delegated access to client environments. This is especially important in MSSP or partner-managed scenarios, where you're managing Microsoft Sentinel across customer subscriptions. You must now decide how to assign roles across tenants.

Choosing the right integration method: GDAP, Lighthouse, or B2B?

Each method has its strengths and limitations. In practice, most MSSPs use a combination of these to meet operational, security, and access control needs.

Azure Lighthouse is required in the following cases:

- **Service principal access across tenants**: Used for automating tasks such as CI/CD deployments and content distribution in Sentinel. It's not compatible with MTO, which uses B2B invitations.
- **Cross-tenant queries in Azure workbooks**: To run Log Analytics queries across tenants, Lighthouse is required. It enables `workspace()` functions to query external workspaces securely.
- **Advanced automation and playbooks**: When an MSSP hosts sensitive playbooks in their own tenant but needs them to interact with customer resources, Lighthouse provides a secure method of cross-tenant access.

Using Lighthouse presents limitations as well. Lighthouse does not grant Microsoft 365 roles. That means if you're relying only on Lighthouse, your SOC analysts may be limited in their investigative or response capabilities within Microsoft 365.

Partner access in multitenant environments remains a complex challenge. There's no single, unified method that covers every need. GDAP, Lighthouse, and B2B each fill different gaps. Understanding their interplay is crucial to building a secure, flexible, and future-ready SOC architecture. As Microsoft continues to evolve its MSSP tooling, particularly with enhancements in Security Copilot, role automation, and playbook workflows, we're optimistic that a more streamlined and consistent model is on the horizon.

Summary

A successful Microsoft Sentinel deployment begins with strong architectural and access foundations. This chapter discussed how to design a scalable, secure environment using Azure management groups, subscriptions, and Log Analytics workspaces. You learned how to structure your hierarchy, enforce governance with policies and diagnostic settings, and choose between existing or dedicated Sentinel subscriptions. With that structure in place, the chapter outlined best practices for designing role-based access using Microsoft Entra ID groups, PIM, and access packages, emphasizing least privilege principles and auditability across both internal and client-facing scenarios.

As you shift into MSSP operations or multitenant environments, this chapter introduced the delegated access models that make scalable service delivery possible: Azure Lighthouse, GDAP, and B2B collaboration. You explored how to deploy foundational RBAC artifacts, build access governance per client, and map team roles to the right Microsoft 365 and Azure RBAC assignments. Whether you're managing internal teams or servicing clients at scale, the tools and models presented here provide the blueprint to enforce Zero Trust, streamline access reviews, and prepare your environment for the future of unified security operations.

In the next chapter, we will discuss Azure Lighthouse as the centerpiece for secure, scalable multitenant access. You will also learn how to onboard customer tenants using ARM templates, map Entra ID groups to SOC-aligned roles, and automate access using PIM and entitlement management.

Get This Book's PDF Version and Exclusive Extras

UNLOCK NOW

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

Note: *Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

7

Customer Onboarding: The Power of Azure Lighthouse

Managing customer environments across multiple tenants can quickly become operationally overwhelming, especially in MSSP or federated enterprise scenarios. Manual role assignments, inconsistent access models, and a lack of centralized visibility introduce risk, inefficiency, and audit complexity. **Azure Lighthouse** addresses these challenges by enabling secure, scalable, and policy-driven delegated access across tenants. It is the mechanism to establish subscription-based trust across tenants and assign roles as if you were a member of that tenant. This chapter focuses on operationalizing Lighthouse as your core onboarding mechanism, helping you streamline access, reduce administrative overhead, and enforce Zero Trust principles from day one.

The chapter covers the following key topics:

- Azure Lighthouse onboarding and best practices
- **Granular delegated admin privileges (GDAP)**
- Entra ID B2B access model
- Client-side onboarding for B2B access

By the end of this chapter, you'll be able to build a scalable, secure, and automation-ready onboarding model that fits both MSSP operations and large enterprise environments with full visibility, governance, and role control from a single management plane.

Let's begin by designing delegated access architecture using Azure Lighthouse.

Technical requirements

To be able to follow along with this chapter, you will need two valid Microsoft 365 tenants with an Azure subscription and Global Administrator and Subscription Owner privileges on both.

Azure Lighthouse onboarding and best practices

In *Chapter 6*, we discussed foundational architecture and secure access for Sentinel deployments for MSSPs. We will now move from getting the platform right to bringing customers on board in a controlled, repeatable way. If the last chapter was about ensuring you can deploy Microsoft Sentinel securely at scale (landing zones, identity and access, connectivity, and governance boundaries), this chapter is about turning that foundation into an operational onboarding motion that consistently produces the same outcome: reliable visibility, predictable cost, and a SOC-ready service.

Most Sentinel onboarding failures are not technical defects; they're process gaps. In MSSP scenarios, inconsistency creates blind spots (missing telemetry), noise (poor signal-to-alert mapping), and friction (unclear ownership across teams). A structured onboarding flow reduces delivery risk, shortens the time to value, and makes your service auditable and defensible, particularly when you need evidence control, separation of duties, and repeatability across multiple customers and tenants.

At this stage, you should already have the *non-negotiables* in place: agreed governance and RBAC model, secure access patterns, workspace strategy, log routing principles, and an initial view of data requirements and retention expectations. With that baseline established, the next step is to execute onboarding in a measurable and scalable way by moving from "*we can deploy Sentinel*" to "*we can onboard customers consistently and run it as a service.*"

Establishing cross-tenant access via Azure Lighthouse

You can use Azure Lighthouse to securely establish cross-tenant access between your organization (e.g., MSSP) and customer environments. We will now create a template and upload it to create the connection between subscriptions. Azure Lighthouse supports the delegation of built-in Azure roles within the Azure RBAC model.



Azure Lighthouse does not support the delegation of Microsoft 365 roles (e.g., Global Administrator, Security Reader, or Compliance Administrator). These roles are managed through Entra ID and must be assigned separately. Use GDAP or Entra ID B2B for collaboration purposes.

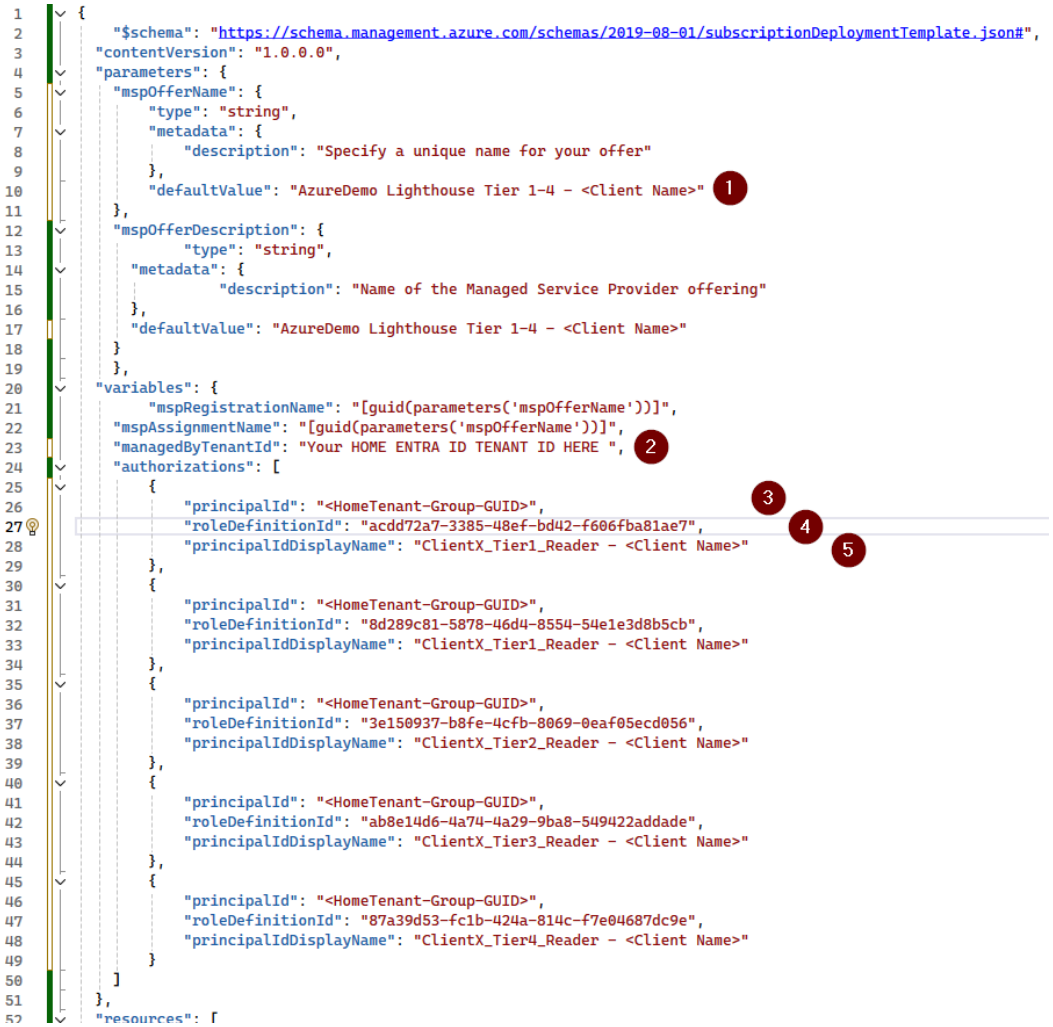
Remember, Lighthouse is not deployed at the tenant level but rather on a subscription or resource group. The word *tenant* is used interchangeably in the industry as subscriptions are bound to a tenant. But technically, it's a subscription.

When planning Lighthouse access, it's important to account for managed-tenant visibility constraints. Customer administrators typically won't see individual MSSP operator access in their tenant logs in a familiar, named-user format, as access is brokered through delegated IAM. As a result, you should explicitly define and document access governance, auditing, and evidence collection.

Reviewing the ARM template

The following ARM template, also available in the GitHub repo (`https://github.com/PacktPublishing/Microsoft-Sentinel-Blueprint`) defines a Microsoft Azure Lighthouse Managed Services offering, allowing the MSSP to delegate role-based access to specific groups across customer tenants. This template illustrates best practices for defining multi-tiered access for SOC roles.

This is what the template should look like:




```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-08-01/subscriptionDeploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "mspOfferName": {
        "type": "string",
        "metadata": {
            "description": "Specify a unique name for your offer"
        },
        "defaultValue": "AzureDemo Lighthouse Tier 1-4 - <Client Name>"
    },
    "mspOfferDescription": {
          "type": "string",
      "metadata": {
              "description": "Name of the Managed Service Provider offering"
      },
      "defaultValue": "AzureDemo Lighthouse Tier 1-4 - <Client Name>"
    }
    },
  "variables": {
        "mspRegistrationName": "[guid(parameters('mspOfferName'))]",
    "mspAssignmentName": "[guid(parameters('mspOfferName'))]",
    "managedByTenantId": "Your HOME ENTRA ID TENANT ID HERE ",
    "authorizations": [
        {
            "principalId": "<HomeTenant-Group-GUID>",
            "roleDefinitionId": "acdd72a7-3385-48ef-bd42-f606fba81ae7",
            "principalIdDisplayName": "ClientX_Tier1_Reader - <Client Name>"
        },
        {
            "principalId": "<HomeTenant-Group-GUID>",
            "roleDefinitionId": "8d289c81-5878-46d4-8554-54e1e3d8b5cb",
            "principalIdDisplayName": "ClientX_Tier1_Reader - <Client Name>"
        },
        {
            "principalId": "<HomeTenant-Group-GUID>",
            "roleDefinitionId": "3e150937-b8fe-4cfb-8069-0eaf05ecd056",
            "principalIdDisplayName": "ClientX_Tier2_Reader - <Client Name>"
        },
        {
            "principalId": "<HomeTenant-Group-GUID>",
            "roleDefinitionId": "ab8e14d6-4a74-4a29-9ba8-549422addade",
            "principalIdDisplayName": "ClientX_Tier3_Reader - <Client Name>"
        },
        {
            "principalId": "<HomeTenant-Group-GUID>",
            "roleDefinitionId": "87a39d53-fc1b-424a-814c-f7e04687dc9e",
            "principalIdDisplayName": "ClientX_Tier4_Reader - <Client Name>"
        }
    ]
  },
  "resources": [
```

Figure 7.1: Onboarding template

Let's look at the key elements of this template:

- `mspOfferName`/`mspOfferDescription`: These parameters define the name and description of the Lighthouse offer. Customize them per client to reflect the delegated access scope (e.g., `Contoso_Tier1_Reader_Access`).
- `managedByTenantId`: This specifies the **Entra ID tenant ID** of the MSSP, which is the organization receiving delegated access. This must be your own tenant ID.

- `principalId`: This refers to the **object ID** of the Entra ID **security group** (**SG**) receiving the delegated role. These groups map to SOC tiers such as the following:
    - Tier 1: Readers
    - Tier 2: Responders
    - Tier 4: Engineers

    For example, use the object ID for `ClientTenantContoso_Tier1_Reader` in the entry where the associated role is `Reader`.

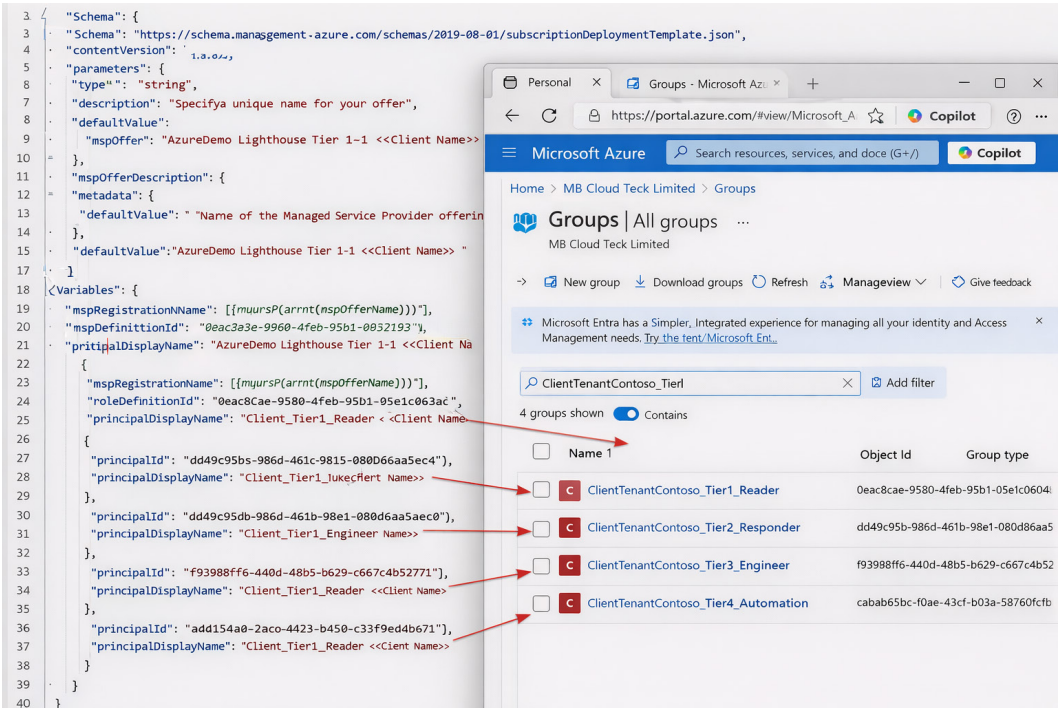


Figure 7.2: Mapping of roles

- `roleDefinitionId`: This field specifies the built-in Azure role ID being delegated. The following are some common examples:

Role	Role Definition ID
Reader	`acdd72a7-3385-48ef-bd42-f606fba81ae7`
Microsoft Sentinel Reader	`b1e8e693-79b9-41d6-a74f-0b621f0f72d4`
Microsoft Sentinel Responder	`3e150937-b8fe-4cfb-8069-0eaf05ecd056`

Role	Role Definition ID
Microsoft Sentinel Contributor	`ab8e14d6-4a74-4a29-9ba8-549422addade`
Logic Apps Contributor	`de139f84-1756-47ae-9be6-808fbbe84772`

Table 7.1: Role definition ID examples



You can find all available roles in the official documentation here: `https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles`.

- `principalIdDisplayName`: This is a human-readable display name that clearly indicates the role and the client context. It improves traceability in the Azure portal and activity logs.

Now that you understand how to use the template, let's discuss how you can keep your assignments clearly structured.

Contextual usage and best practices

You can include multiple role assignments by duplicating the block structure within the ARM template. If multiple roles, such as Reader and Sentinel Reader, are assigned to the same group, the `principalId` value remains the same across both entries. This design enables granular access control based on SOC tiers or personas, including auditor, responder, and automation engineer.

```
18        }
19        },
20    "variables": {
21          "mspRegistrationName": "[guid(parameters('mspOfferName'))]",
22        "mspAssignmentName": "[guid(parameters('mspOfferName'))]",
23        "managedByTenantId": "Your HOME ENTRA ID TENANT ID HERE ",
24        "authorizations": [
25            {
26                "principalId": "<HomeTenant-Group-GUID>",
27                "roleDefinitionId": "acdd72a7-3385-48ef-bd42-f606fba81ae7",
28                "principalIdDisplayName": "ClientX_Tier1_Reader - <Client Name>"
29            },
30            {
31                "principalId": "<HomeTenant-Group-GUID>",
32                "roleDefinitionId": "8d289c81-5878-46d4-8554-54e1e3d8b5cb",
33                "principalIdDisplayName": "ClientX_Tier1_Reader - <Client Name>"
34            },
35            {
36                "principalId": "<HomeTenant-Group-GUID>",
37                "roleDefinitionId": "3e150937-b8fe-4cfb-8069-0eaf05ecd056",
38                "principalIdDisplayName": "ClientX_Tier2_Reader - <Client Name>"
39            },
```

Figure 7.3: Role assignments

As you can see in *Figure 7.3*, two roles are assigned to the same group.

Deploying an Azure Lighthouse template

Once the Azure Lighthouse ARM template has been created and customized, the next step is to deploy it from the client tenant. This can be done either directly by the client's Azure administrator, using the template file or a Managed Services offering, published to the Azure Marketplace, or shared privately:

1. Go to `portal.azure.com` and, using the credentials for the client tenant, sign in.
2. From the left-hand navigation, navigate to **Azure Lighthouse** and click **View service provider offers**.

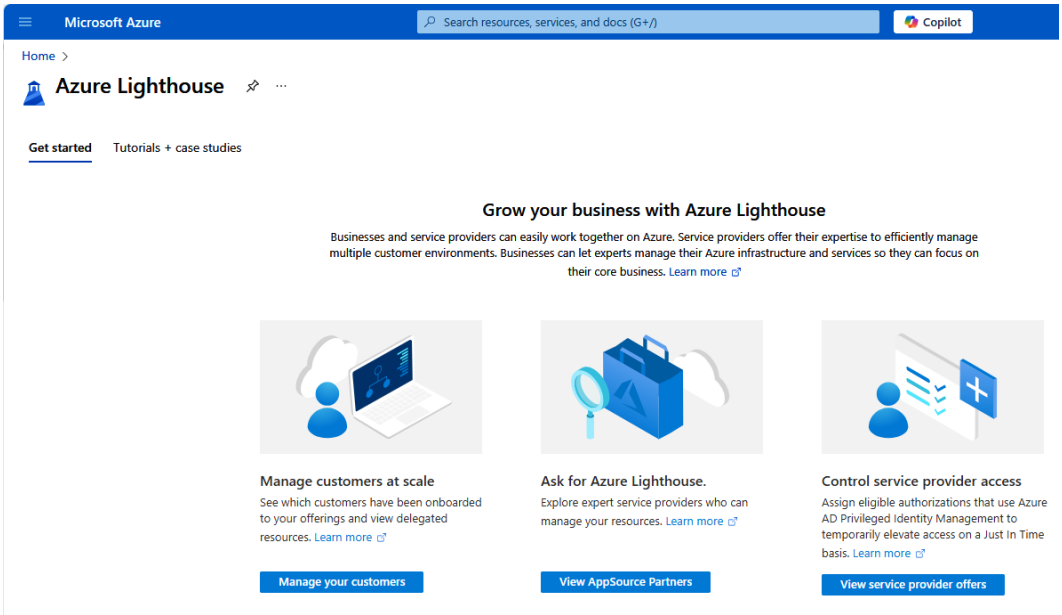


Figure 7.4: Azure Lighthouse in Azure

3. Click **+Add offer** and select **Add via template** from the drop-down list.

4. Upload the customized ARM template file (JSON).

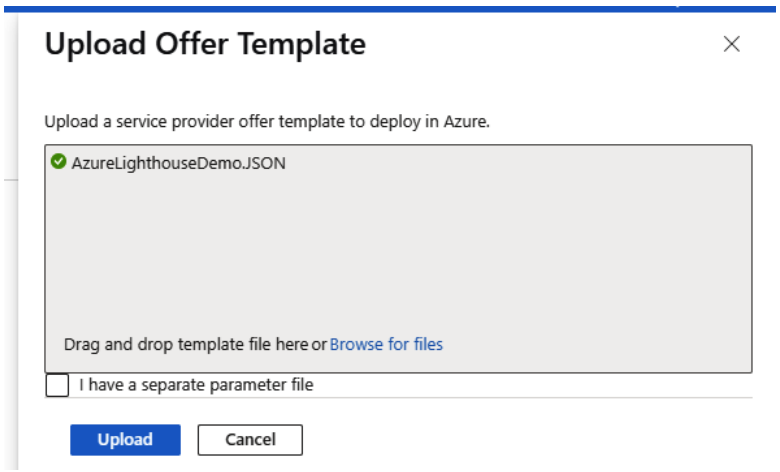


Figure 7.5: Upload JSON file

5. Confirm the details populated from the template. Then, run the automatic validation and click **Deploy**.

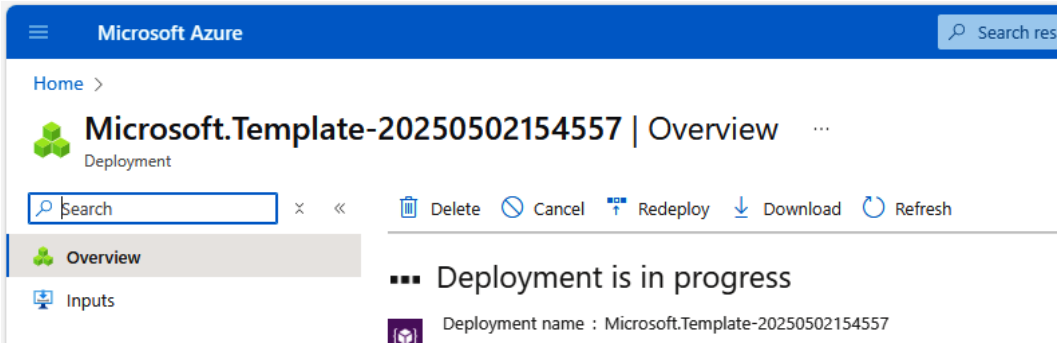


Figure 7.6: Deployment of template in progress

6. After the deployment, navigate back to **Service provider offers** to view assigned roles, as shown in *Figure 7.7*.

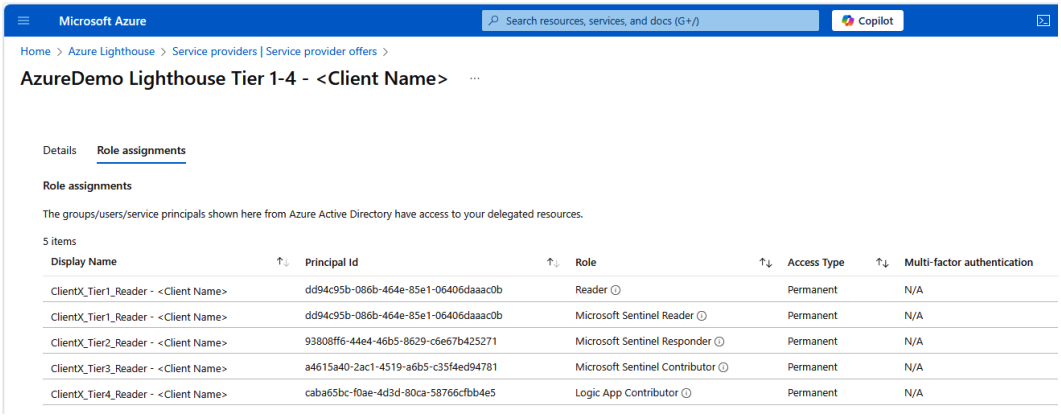


Figure 7.7: Lighthouse role assignments

You will see the delegated roles listed (e.g., Reader or Sentinel Responder). You can manually add access to additional subscriptions by clicking the + icon on the overview screen.



It may take up to 15 minutes for the client's subscription ID to appear in your subscription list (from the MSSP tenant view). Also, ensure you have activated PIM access to the SGs that were referenced in the ARM template. These groups control the delegated roles. Use the appropriate dropdown in the Azure portal to verify your active role assignments.

To validate the presence of the newly set up connection, navigate to settings in Azure and take a look at your newly available delegated subscriptions.

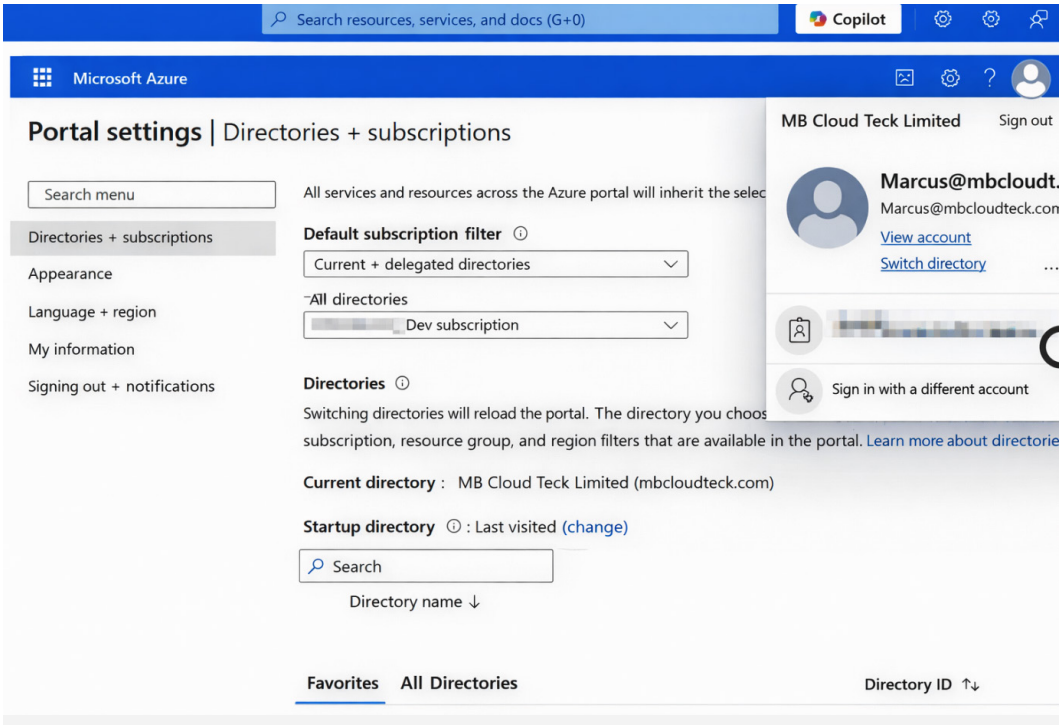


Figure 7.8: Verify directories and subscriptions under Portal settings

In addition to manual template deployment, Azure Lighthouse supports the following alternative deployment options:

- **Azure Marketplace:** Publish a managed offer that clients can accept with minimal interaction
- **REST API/PowerShell/CLI:** Automate deployments at scale using programmatic methods
- **Bicep:** Deploy using declarative, version-controlled infrastructure-as-code templates.

These methods will not be covered in this book, due to their complexity and lack of flexibility—maybe in our second edition!

Granular delegated admin privileges (GDAP) connection

GDAP allows **Microsoft Partners** to implement refined, role-specific administrative access to customer tenants. Unlike traditional **delegated admin privileges** (**DAP**), GDAP enables scoped access based on Microsoft Entra roles. Remember that we are now discussing Microsoft 365 roles and not Azure roles. They sit on two sides of the same fence.



While it is technically possible to access customer Azure resources via GDAP using Admin Agent, this grants Owner permissions but violates least-privilege principles. Instead, we recommend using Azure Lighthouse in combination with GDAP or B2B collaboration for secure and compliant access. Learn more here: `https://learn.microsoft.com/en-us/partner-center/customers/gdap-faq`.

To set up GDAP, follow these steps.

Step 1: Creating a new admin relationship

1. Navigate to `https://partner.microsoft.com/` and sign in using an account with the Admin Agent role in Microsoft Partner Center.

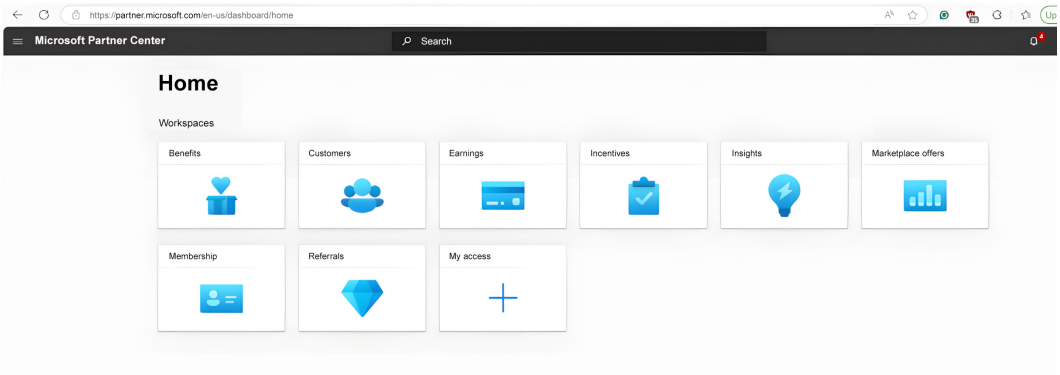


Figure 7.9: Partner Center menu

2. To request a new admin relationship, go to **Customers**, select the customer, and click **Request admin relationship**. You must have the Admin Agent role to view or use this feature.

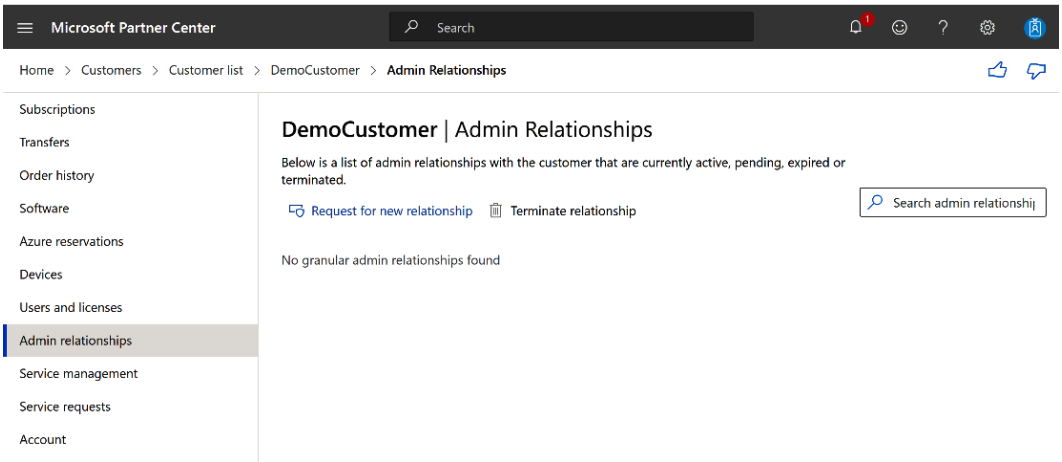


Figure 7.10: GDAP relationships dashboard

3. Now, you can configure the new admin relationship. Fill in the fields as follows:
    - **Admin relationship name**: Provide a unique, descriptive name (e.g., `Contoso_SecurityOps_Tier1`).
    - **Duration in days**: Define the validity period. GDAP roles expire automatically after this period.
4. Assign Microsoft Entra roles by clicking the **Select Microsoft Entra roles** tab, which opens to a side panel with role options. Choose only the roles required for your operational tasks.

    Recommended roles for security operations include the following:

    - Security Administrator
    - Security Operator
    - Security Reader
    - Global Reader
    - Reports Reader
    - Intune Administrator
    - Help Desk Administrator



Lessons from the field

We strongly discourage assigning Global Administrator via GDAP. This role should not be required by a service provider if they're acting in coordination with a client following least privilege.

5. Once you have decided on and selected roles, finalize the request. This will then create a template to send to your client with all the information and a link.

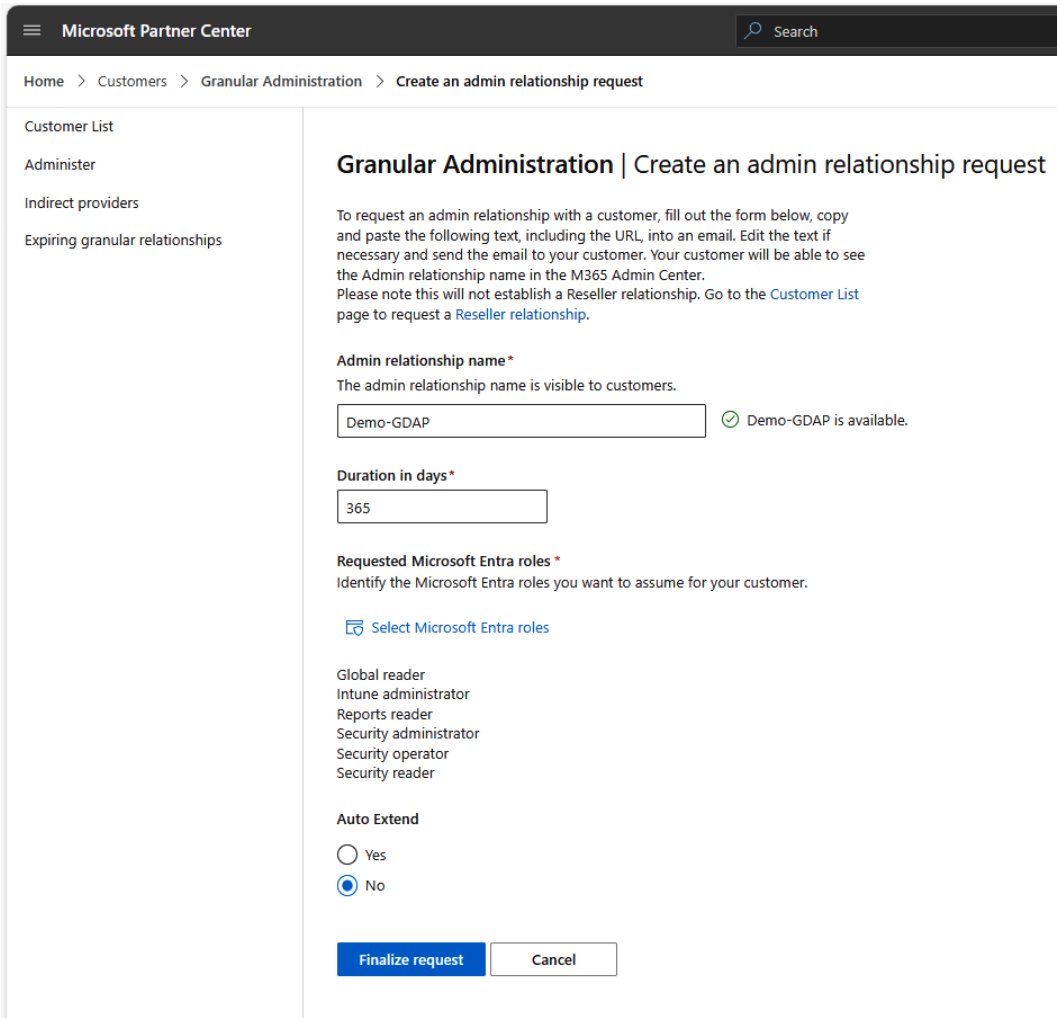


Figure 7.11: Creation of a GDAP relationship

Step 2: Sending and finalizing the GDAP request

After completing the configuration, a GDAP invitation link will be generated. This link is included in an auto-generated email, which you can copy and send to your client, as shown in *Figure 7.12*.

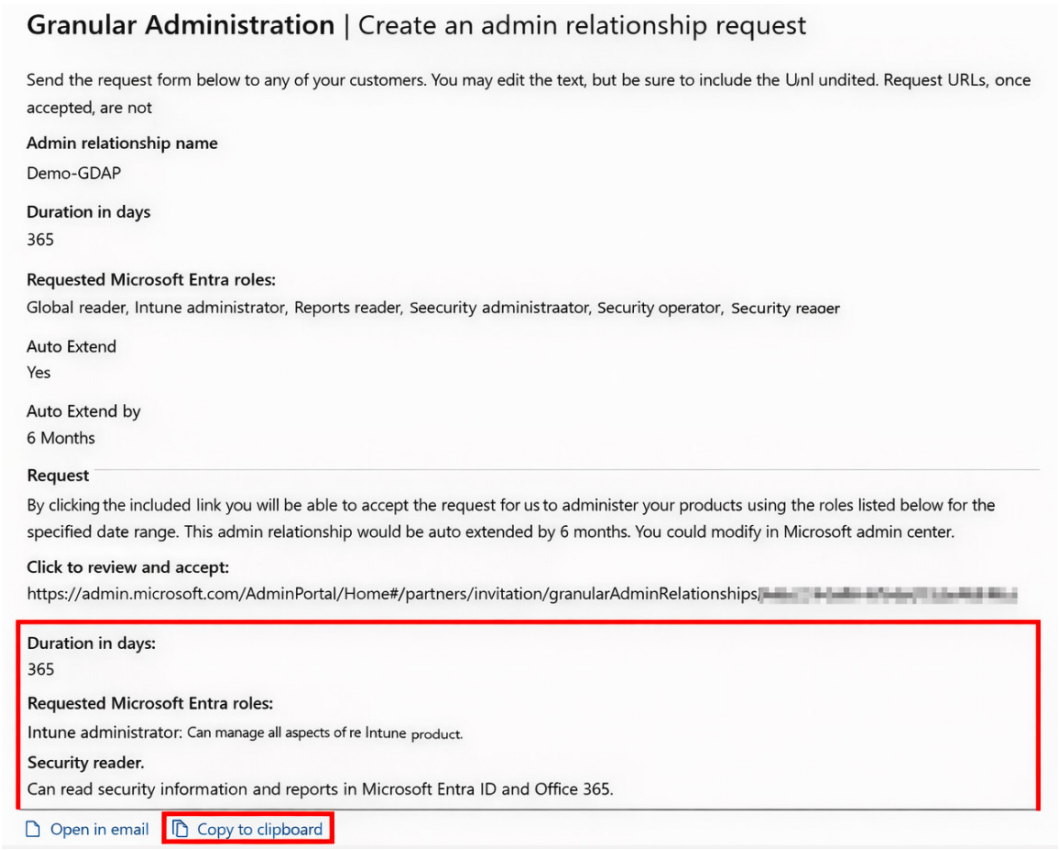


Figure 7.12: Customer to review and accept the relationship invite

A Global Administrator in the customer tenant must review and approve the request. Once the GDAP relationship is approved, the next step is to map the roles to Azure AD SGs.

Step 3: Mapping GDAP roles to Azure AD security groups

This step forms the connection between the approved roles and the home tenant's access methodology.

Follow these steps to map the GDAP roles:

1. Return to Partner Center, select the customer, and click **Administer**.
2. Navigate to **Security groups** and begin mapping roles. Use the SGs you created earlier (e.g., `ClientTenantContoso_Tier1_Reader`) and assign relevant Entra roles.

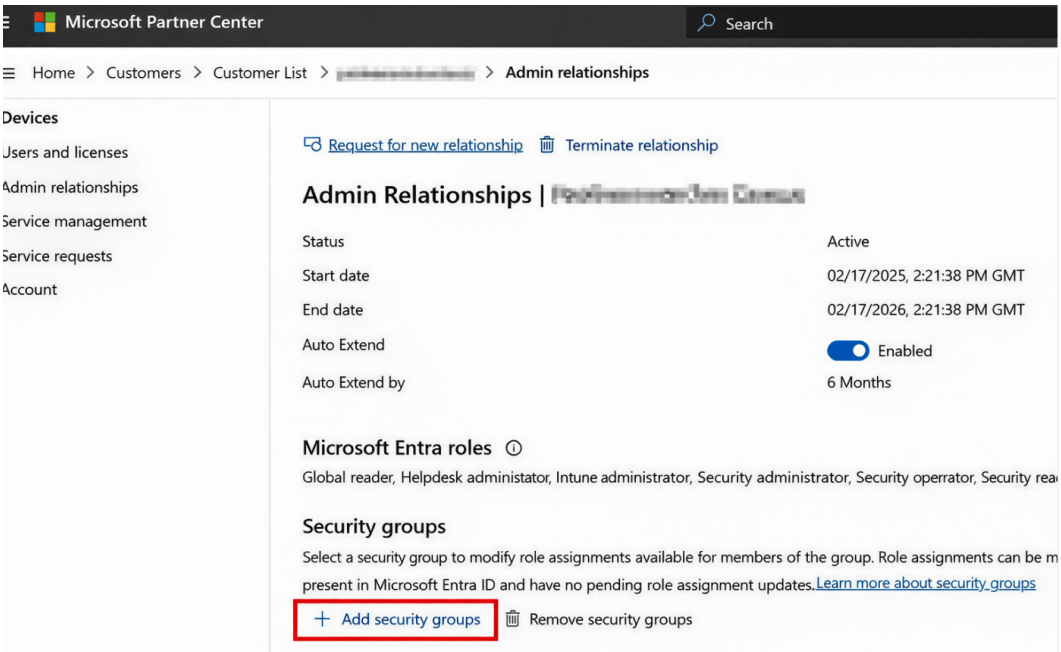


Figure 7.13: Map SGs with Entra ID roles

For example, assign Global Reader and Security Reader roles to the `ClientTenantContoso_Tier1_Reader` group. This setup allows users from your home organization to PIM into the group and assume the GDAP-assigned role within the client tenant.

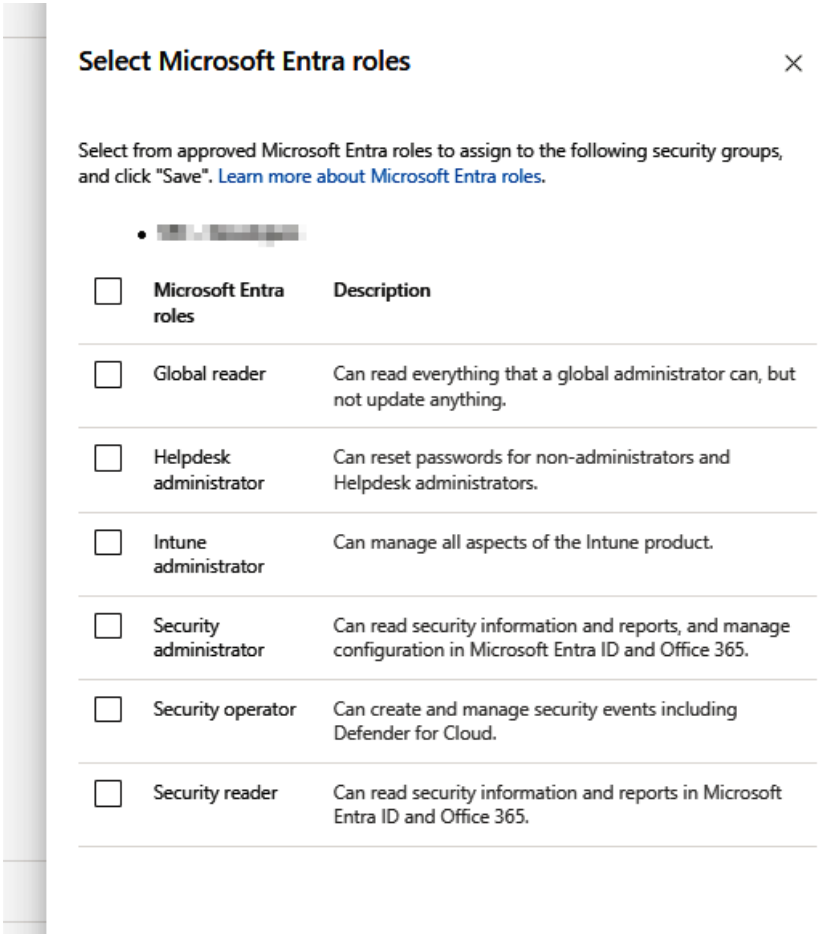


Figure 7.14: Role options

This method enables RBAC through group membership, supporting least-privilege access and improved auditing. Partner Center does not natively support this level of mapping, so manual group-to-role assignment is required.

Step 4: Accessing the client environment via GDAP

To access client resources, follow these steps:

1. Go to **Partner Center | Customers**.
2. Select the desired customer and then click **Administer**.
3. Choose the appropriate portal (e.g., Microsoft 365 or Azure) to access the client environment with the roles granted via GDAP.

Now that you've learned how to use GDAP at scale for your clients in a granular, least-privilege way, let's look deeper into how to map and access sub or client tenants via a B2B connection.

Entra ID B2B access model

This section moves on to the Entra B2B model that can complement the previously explained GDAP and Lighthouse access. At this stage, the target outcome is to enable an MSSP (home tenant) to operate inside a client tenant securely and at scale, using Entra B2B, cross-tenant access trust, identity governance, and PIM, while keeping access auditable, role based, and time-bound.

For maximum flexibility and enterprise-grade security alignment, the Entra ID B2B collaboration model is the recommended approach for cross-tenant access. This model offers robust governance, granular access control, and deep integration with Microsoft security tools such as Microsoft Sentinel and Microsoft Defender.



B2B access comes with a cost of ownership, including licensing and administrative overhead, which must be considered during planning.

We will now discuss the key steps required to implement the B2B model. These steps will be explained in detail in subsequent sections to ensure both client and home tenant teams have a consistent and standardized process:

1. Invite an initial guest administrator:
    - The client tenant invites the first guest user from the home tenant
    - Assign appropriate administrative roles to this guest to enable initial configuration tasks

2. Establish organizational trust:
    - Use the guest account to configure the B2B trust
    - Establish directory-to-directory collaboration between the home tenant and client tenant
3. Create access catalogs and packages:
    - Set up an access catalog and define at least five **access packages (APs)**
    - Each package should correspond to an SG, mapped to distinct access levels
4. Define eligibility for guest users:
    - Configure Entitlement Management to control which home tenant users are eligible for access
    - Ensure only pre-approved guest users can request or be assigned roles
5. Assign group owners and approvers:
    - Assign group owners and approvers within the home tenant to oversee access requests, approval workflows, and lifecycle operations
6. Configure access for Microsoft Sentinel:
    - Link APs to PIM-eligible groups
    - Ensure roles such as Sentinel Reader, Responder, and Contributor are correctly scoped
7. Configure access for Microsoft Defender:
    - Align APs with unified RBAC roles across the Defender suite, including Defender for Endpoint, Defender for Identity, and Microsoft Defender XDR
8. Test and document the configuration:
    - Perform full testing to validate end-to-end access and role assignments
    - Document each configuration step and outcome to support future audits and troubleshooting

Mandatory requirements

To implement B2B access securely and effectively, the following prerequisites must be in place. They are required to make various features and elements of the solution active and work:

- Microsoft Entra ID P2 licenses are required to support entitlement management, access reviews, and PIM
- Defender unified RBAC must be enabled for Microsoft Defender for Endpoint, Microsoft Defender for Identity, and Microsoft Defender XDR (formerly Microsoft 365 Defender)
- A guest user from the home tenant must be invited and accepted into the client tenant and assigned the necessary administrative rights within the Defender XDR portal
- A valid email for a client-side approver (ideally Global Administrator) is required to own the first AP

You should also take a few key considerations into account during this process. Group management must be tightly controlled. Since this model depends on group-based PIM roles, improper group editing rights can result in excessive privilege or misconfiguration. Delegate sparingly and monitor changes closely. Also, ensure that access management aligns with insider risk policies. This enforces consistent governance across all collaborative and administrative touchpoints.

We will start with the client-side tenant, with the assumption that the client has accepted a task and is inviting you on board.

Client-side onboarding for B2B access

To begin the B2B onboarding process, the client tenant must first invite the administrator responsible for setting up the B2B connection. This can be done by either the client admin inviting and granting the necessary roles to the guest user from the MSSP (home tenant) or the client setting up the environment themselves, following the same steps outlined.

Step 1: Configuring external identity and cross-tenant access

Before any resources can be accessed or managed across tenants, a secure identity trust must be established between the client tenant and the home tenant. This step lays the foundation for cross-tenant collaboration by defining how identities are recognized, trusted, and governed, ensuring that access is both intentional and controlled before proceeding with delegated administration or service onboarding.

1. Sign in to the client tenant and navigate to the Microsoft Entra admin center.

2. Go to **External Identities | Cross-tenant access settings**.

   To add and configure trust settings for the home tenant, click **+ Add organization** and enter the `<home tenant name>.onmicrosoft.com` domain.

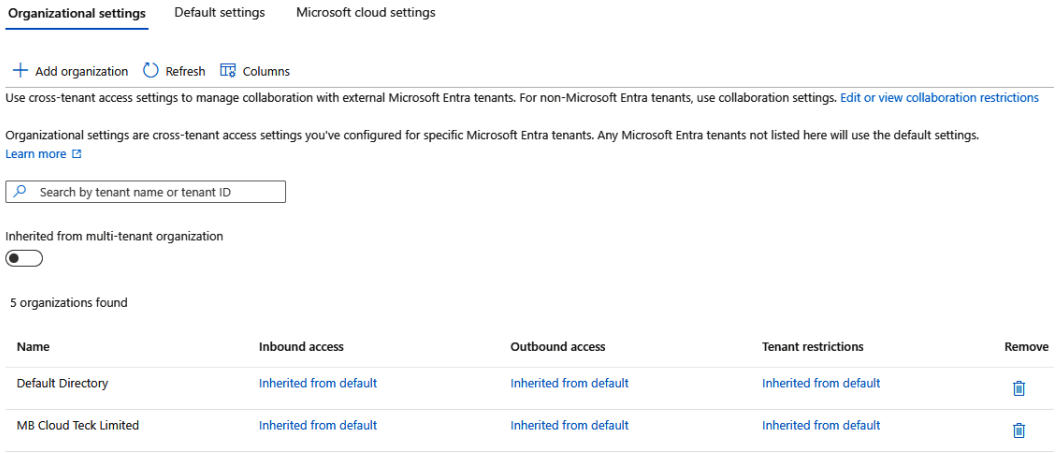


Figure 7.15: Add trust settings for the home tenant

3. For **Inbound access**, select **Inherited from default**, then choose **Customize settings**.

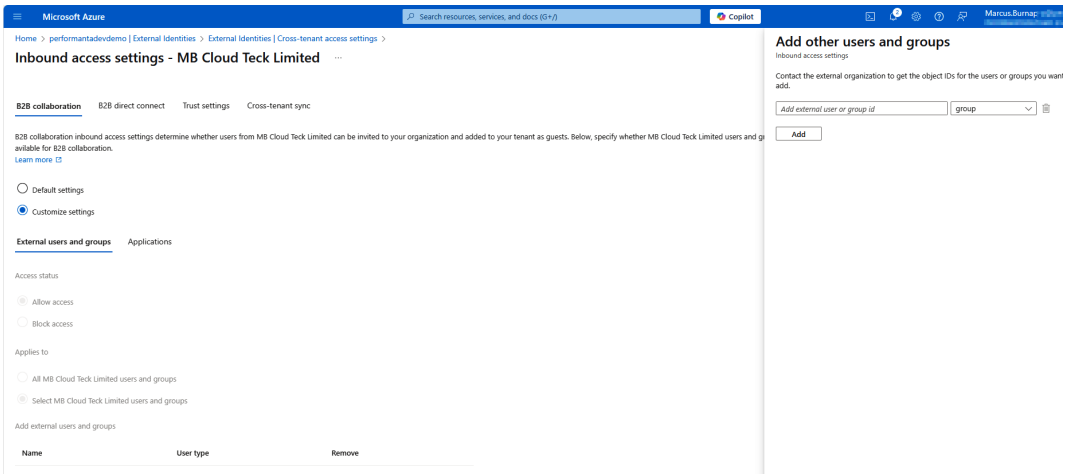


Figure 7.16: Customize settings for inbound access

4. Enable trust for MFA from the home tenant. This ensures the client tenant will honor MFA claims issued by the MSSP tenant.

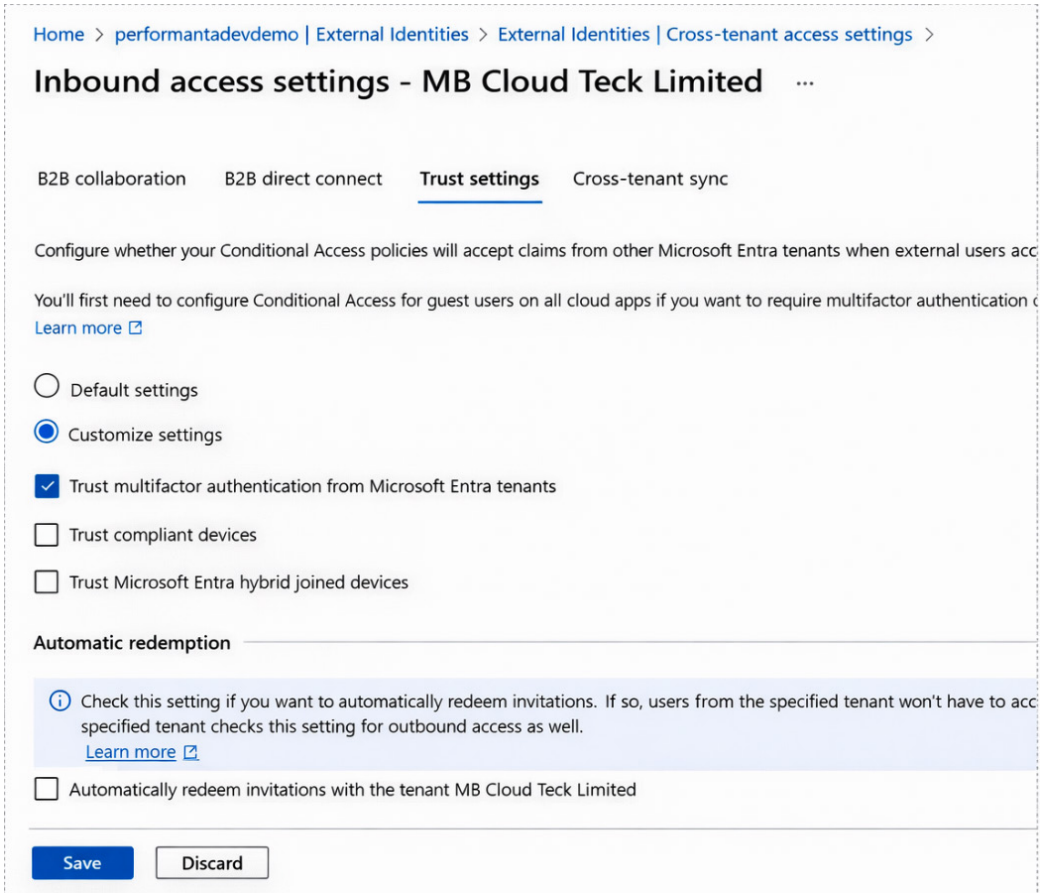


Figure 7.17: Enable MFA trust

Now that the tenant requirements are in place, the next step is to ensure users can traverse through your tenant Conditional Access requirements.

Step 2: Creating an authentication context and Conditional Access policy

The next step will prepare us for setting up Conditional Access later in the chapter. To support passwordless MFA and Conditional Access for elevated roles, you must follow these steps:

1. Navigate to the Microsoft Entra admin center | **Protection** | **Conditional Access** | **Authentication Contexts**.
2. Click **+ New authentication context** and configure the context with the following details:
    - Name the context clearly (e.g., `HomeTenantName_AuthContext`)
    - Uncheck **Publish to apps**
3. Click **Save**.

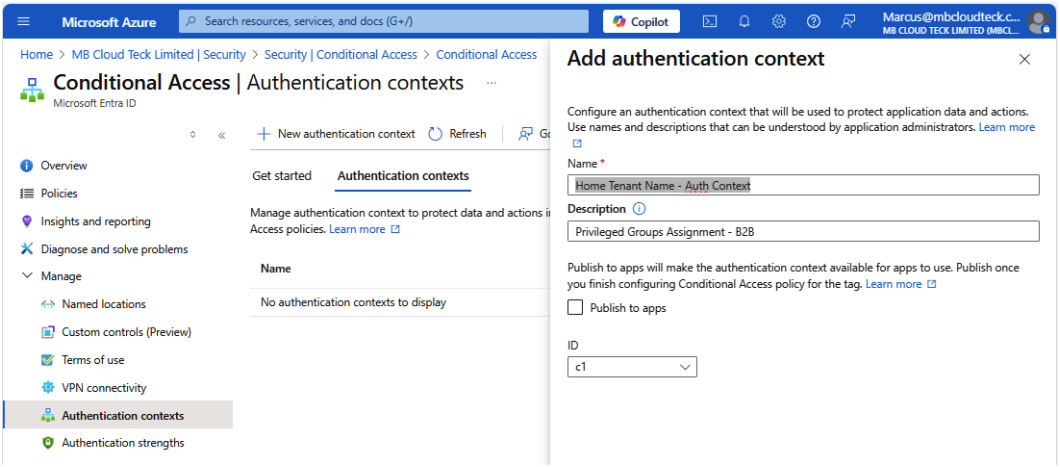


Figure 7.18: Configure the authentication context



Creating authentication contexts requires either a Global Administrator or Conditional Access Administrator role. This context will later be referenced in a Conditional Access policy to enforce MFA or other security requirements for specific groups or role elevation scenarios.

Step 3: Inviting initial B2B MSSP sponsors

Next, invite key users from the MSSP (home tenant) who will act as external sponsors and approve access requests. Managed services managers, analyst managers, engineering managers, and Microsoft consultants are typically senior stakeholders responsible for managing operational access.

Assign usage locations to each invited user. Use the Microsoft Entra admin center | **Users** | **New Guest User** to send invitations.

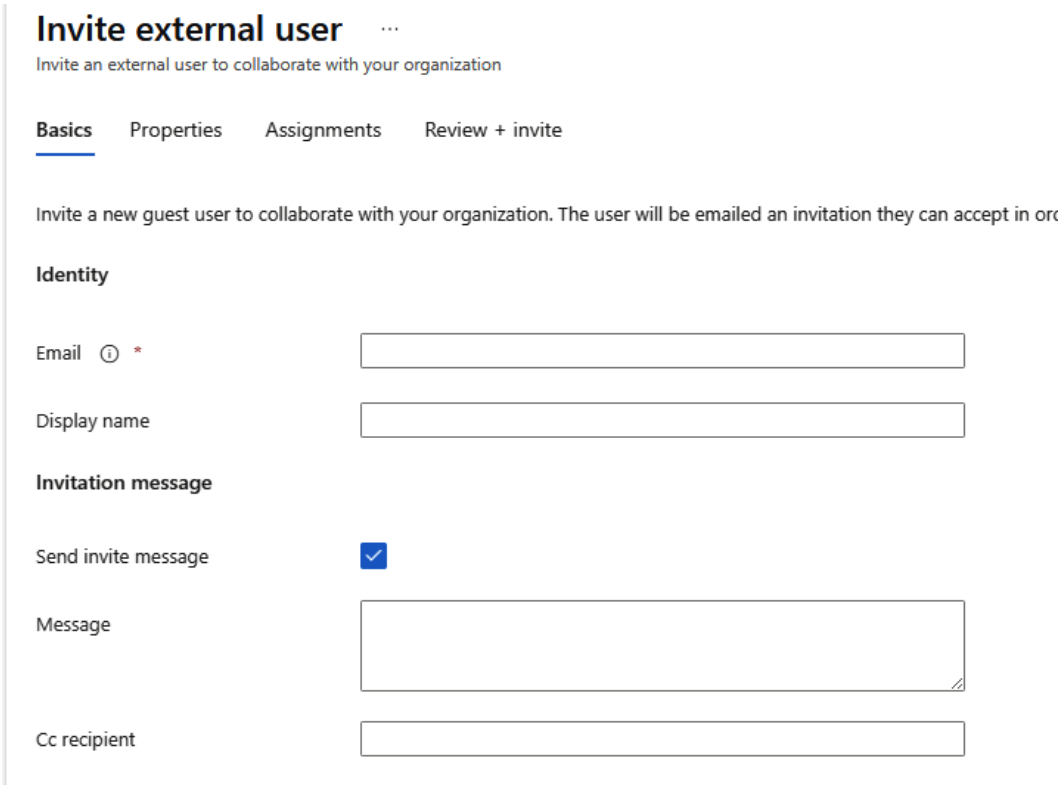


Figure 7.19: Invite external users

Step 4: Adding the home tenant as a connected organization

With cross-tenant trust established and access policies defined, the next step is to formally register the home tenant within the client's identity governance framework. Adding the home tenant as a connected organization enables Microsoft Entra to apply governance controls such as access reviews, entitlement management, and lifecycle policies ensuring that delegated access remains auditable, time-bound, and aligned with the client's identity governance model.

1. Navigate to **Identity Governance | Connected Organizations**. Click **+ New connected organization**.

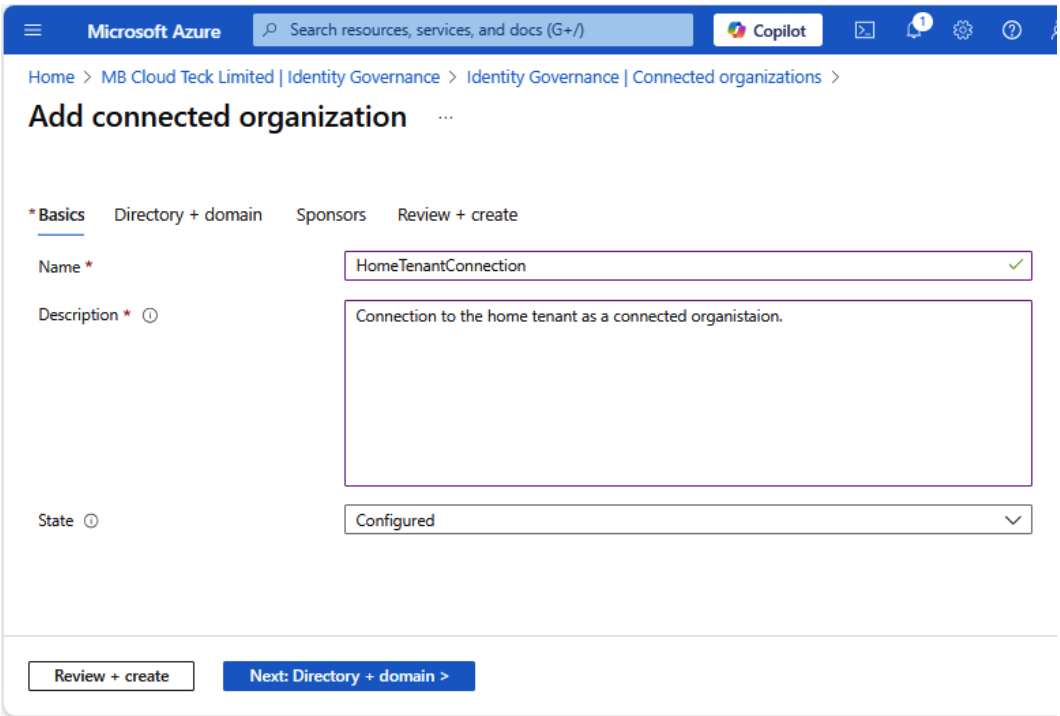


Figure 7.20: Add home tenant as a connected organization

2. Configure the connection by filling in the fields shown in *Figure 7.20*:

    - **Name**: Use your MSSP or home organization name (e.g., `Contoso MSSP`)
    - **Domain**: Enter the MSSP domain (e.g., `contoso.com`)

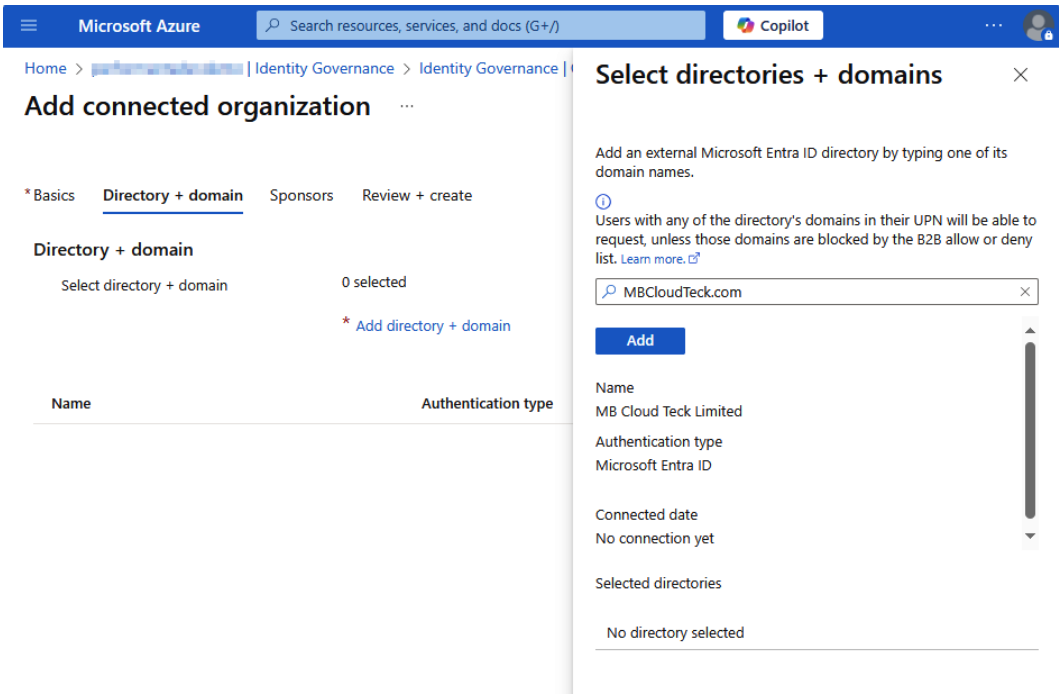


Figure 7.21: Configure the connection

3. Assign the key stakeholder who will approve the connection on both sides:

    - **Internal sponsor**: Client-side approvers (e.g., client managers)
    - **External sponsor**: MSSP users you just invited (from *Step 3*)

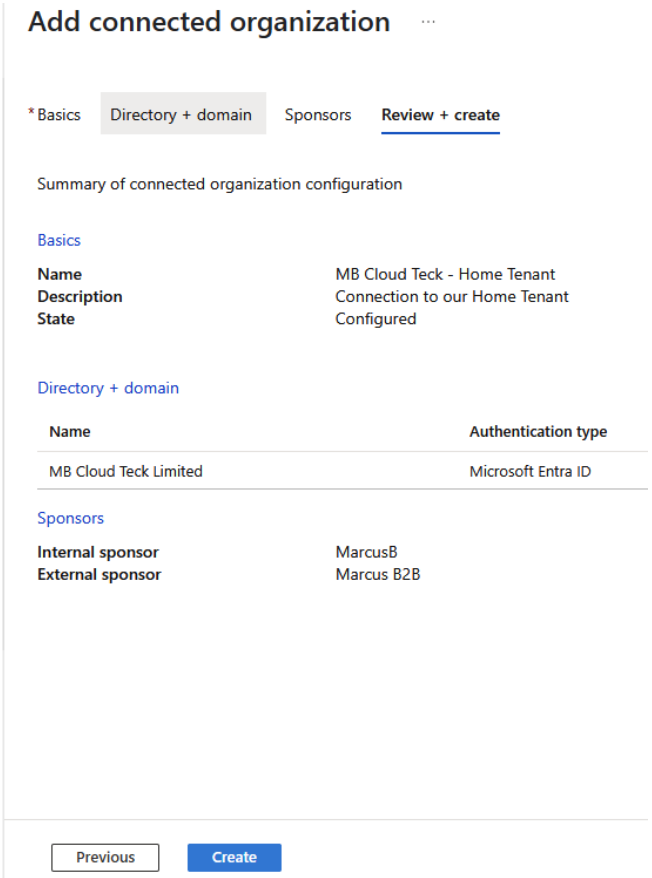


Figure 7.22: Assign internal and external sponsors

4. Click **Review | Create** to complete the setup.

At this point, the client tenant is fully configured to trust MFA claims from the MSSP (home tenant), apply authentication context to enforce secure access policies, and allow B2B users from the home tenant to elevate into privileged roles using PIM and APs, while maintaining full governance.

This setup ensures secure collaboration, RBAC, and compliance with enterprise Zero Trust principles. Since you have created the setup for the platform, now you must govern the users who will flow through the previous setup. For this, we should use a secure and scalable approach.

Step 5: Creating access package security groups

We're now going to create a few sets of SGs. Just as we set things up in a granular way previously, we now need to establish a scalable structure on the client tenant in this B2B scenario. We'll create SGs that will govern APs, grant membership into other groups, and act as PIM entitlement groups for varying access requirements.

For demonstration purposes, we'll follow two logical boundaries: one group for Sentinel and Azure membership and another one for URBAC. We are including the URBAC listing not only as a best practice for segregating access but also as a strong indication that Sentinel components and Azure-dependent fabric resources (e.g., Logic Apps) could be brought under URBAC scope soon. In that case, you would simply assign a URBAC role to your Sentinel group.

Let's create groups to govern APs within Entra ID:

1. Create an Access Package housing for the sub-teams within. Sub teams will operate with access to a specific access package, one or many, allowing you to craft modular access packages, for regional or larger teams. The naming convention here we have used is tenant name, AP, SG, and team role:
    - `MBCloudTeck AP SG SecurityAudit`
    - `MBCloudTeck AP SG SecurityResponders`
    - `MBCloudTeck AP SG SecurityEngineers`
    - `MBCloudTeck AP SG AutomationTeam`

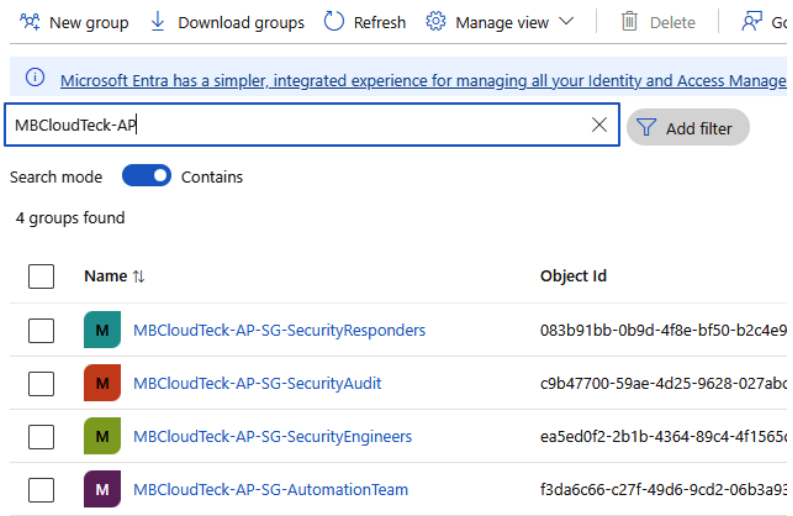


Figure 7.23: Create groups for APs

The previously mentioned groups will govern access to Sentinel and any required Azure resources, respectively:

- `MBCloudTeck Sentinel CloudSecurity_Tier1_Reader`
- `MBCloudTeck Sentinel CloudSecurity_Tier2_Responder`
- `MBCloudTeck Sentinel CloudSecurity_Tier3_Engineer`
- `MBCloudTeck Sentinel CloudSecurity_Tier4_Automation`

2. We will also need to do this for Microsoft 365/Defender access. In Entra, create the corresponding groups for Entra Global/URBAC configuration:

- `MBCloudTeck URBAC CloudSecurity_Tier1_Reader`
- `MBCloudTeck URBAC CloudSecurity_Tier2_Responder`
- `MBCloudTeck URBAC CloudSecurity_Tier3_Engineer`
- `MBCloudTeck URBAC CloudSecurity_Tier4_Automation`

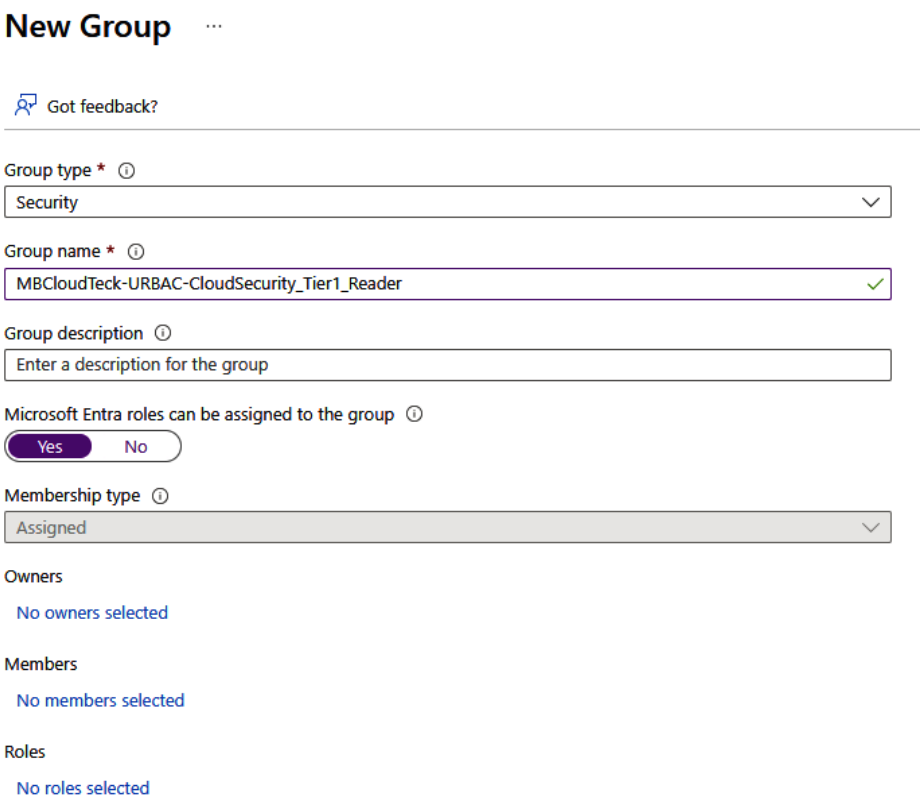


Figure 7.24: Create group

These groups are simplified examples, and in practice, they could be made more granular. For instance, you can target specific products in the Microsoft Defender stack with access defined at either the read or write level. That's the beauty of the new capabilities in the URBAC portal, which we'll explore later.

3. In the groups that we have just created, both `Sentinel` and `URBAC`, amend the **Member** settings.

   Within Entra ID, find the relevant group and navigate to **Privileged Identity Management | Settings**.

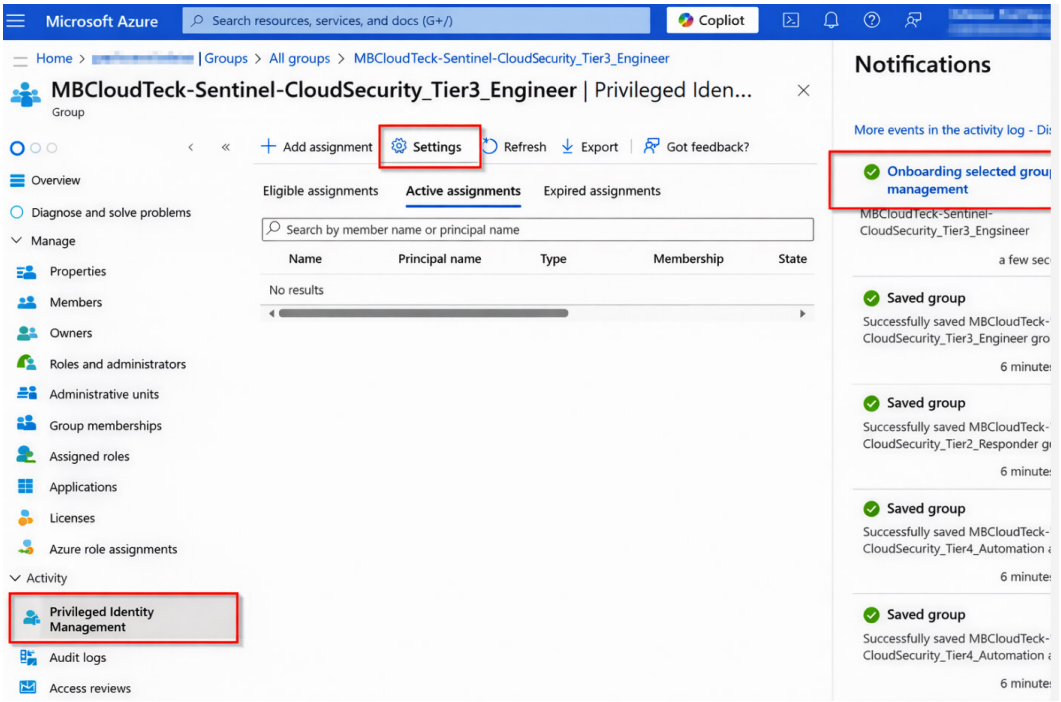


Figure 7.25: Member settings menu

4. Configure PIM configuration for the `Sentinel` and `URBAC` groups as follows:

Setting	State
Activation maximum duration (hours)	8 hour(s)
On activation, require	Authentication context: Home Tenant Name Auth Context
Require justification on activation	Yes
Require ticket information on activation	Yes
Require approval to activate	No
Approvers	None

Table 7.2: PIM group settings

5. Configure the assignment settings as follows:

Setting	State
Allow permanent eligible assignment	Yes
Expire eligible assignments after	Not specified
Allow permanent active assignment	No
Expire active assignments after	15 day(s)
Require Azure MFA on active assignment	Yes
Require justification on active assignment	Yes

Table 7.3: PIM group assignment settings

6. Make the AP membership group eligible. Users in the `MBCloudTeck AP SG SecurityEngineers` AP will have the option to PIM into both `MBCloudTeck Sentinel CloudSecurity_Tier3_Engineer` and `MBCloudTeck URBAC CloudSecurity_Tier3_Engineer`.

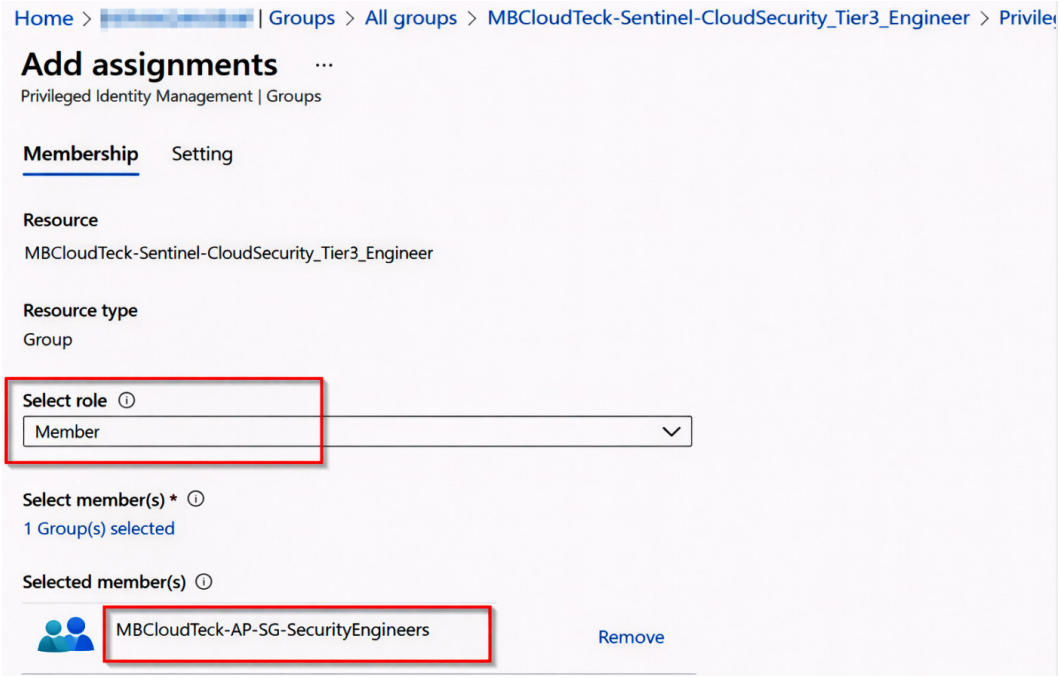


Figure 7.26: Add members to an AP

With our PIM settings in place, let's start adding actual privilege assignments.

Step 6: Adding role assignments to the groups

Let's assign the appropriate roles to the SGs we've created.

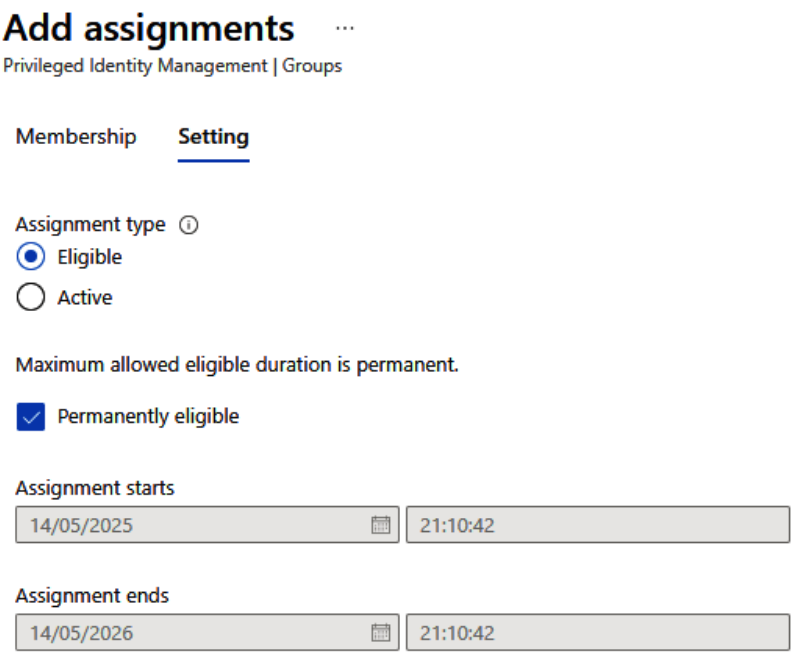


Figure 7.27: Add assignments

Azure/Microsoft Sentinel role assignments

Let's begin with the Azure resources:

1. Navigate to the Microsoft Entra admin center | **Privileged Identity Management** and select **Azure resources**.

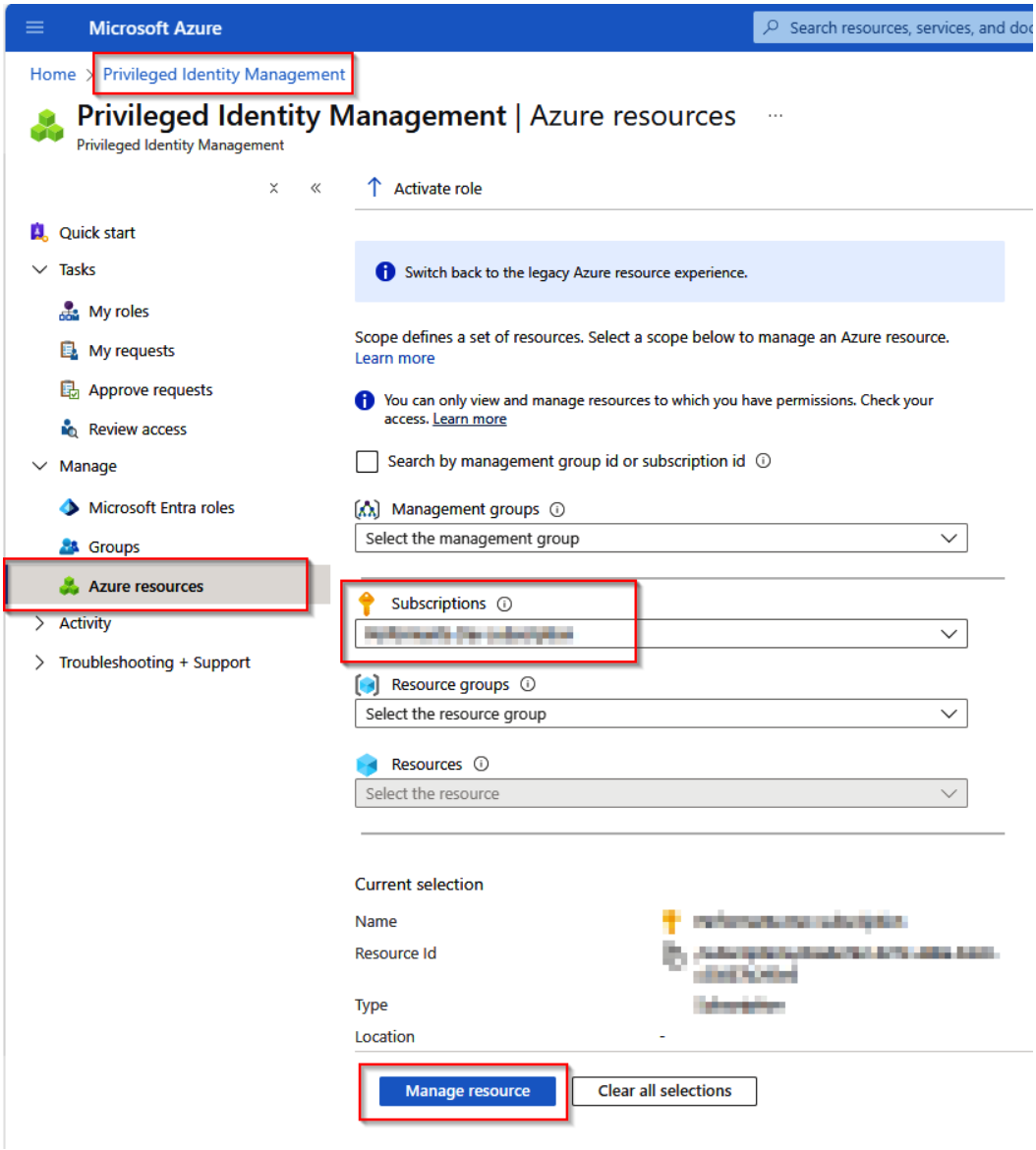


Figure 7.28: Adjust resource type filters

If the Sentinel subscription is not visible, click **Discover Resources** and adjust the **Resource Type** filter to include the relevant subscription and resource group.

Step 1: Making roles available via the authentication context

Before assigning them, we need to ensure the roles are available to the appropriate authentication context and configured to allow permanent active assignments:

1. To do this, locate the role you want to assign (e.g., Reader) and edit the role settings.

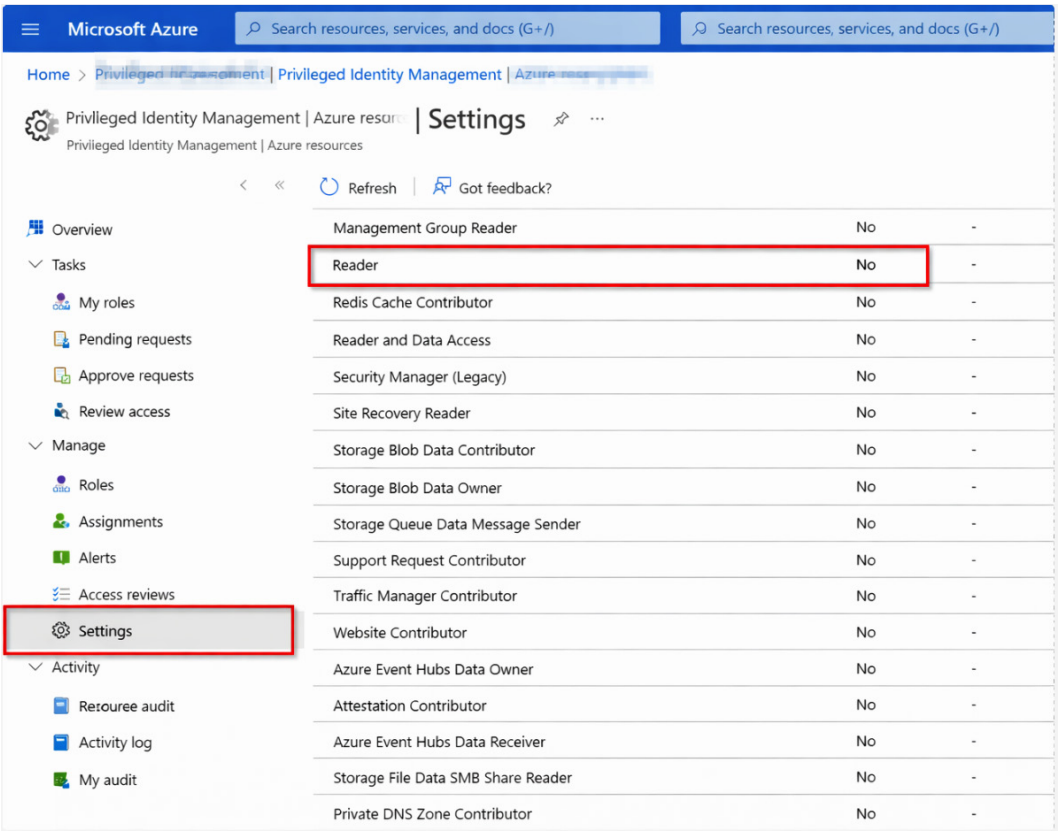


Figure 7.29: Locate the role you want to assign Reader

2. Select the previously created authentication context, which is listed in the dropdown by default.

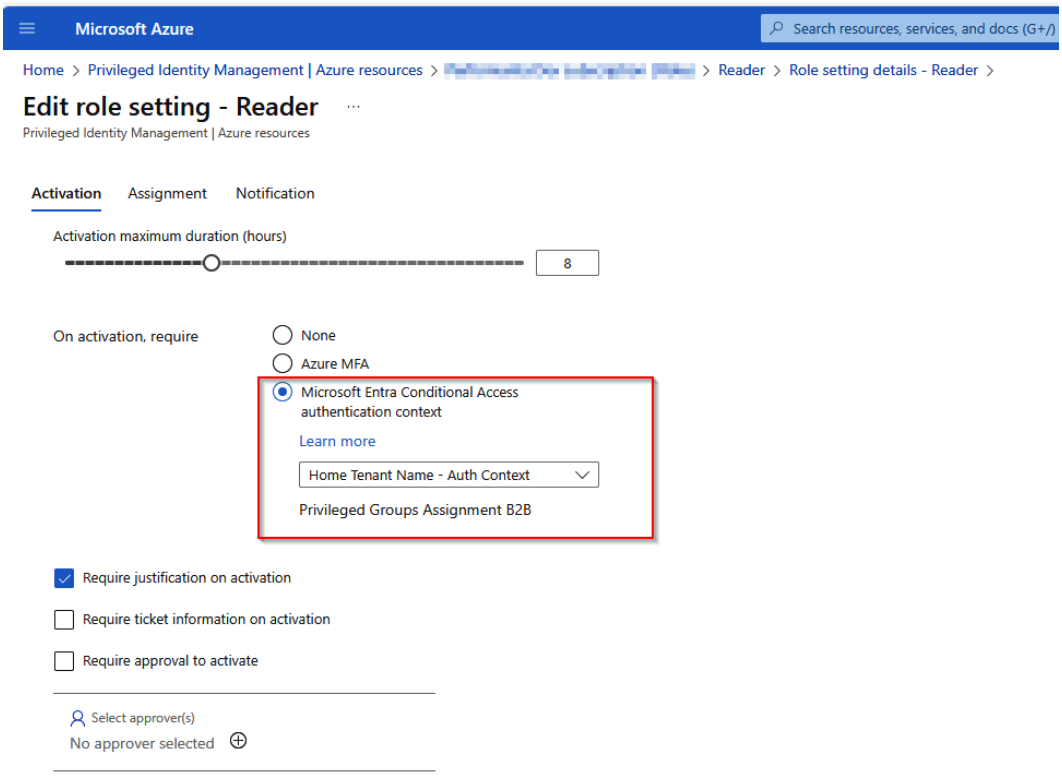


Figure 7.30: Edit role settings

3. Then, enable permanent assignment.

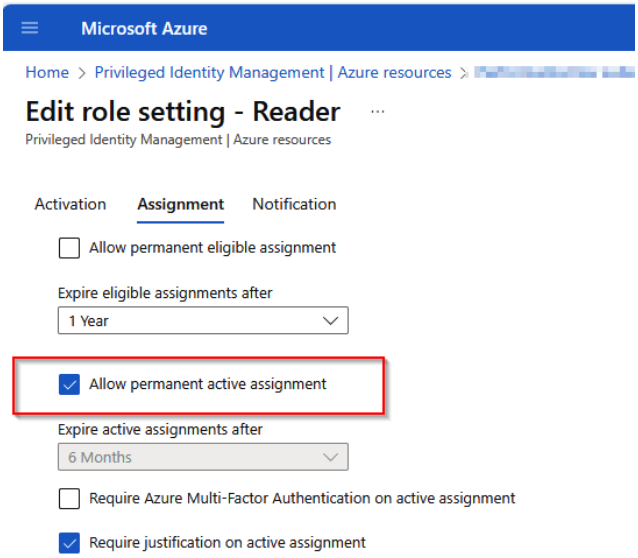


Figure 7.31: Allow permanent assignments

Step 2: Repeating the process for required roles

Based on the example personas, repeat the preceding process for the following roles:

- Reader
- Contributor
- Microsoft Sentinel Reader
- Microsoft Sentinel Responder
- Microsoft Sentinel Contributor
- Logic Apps Contributor

It's recommended to deploy Microsoft Sentinel in a dedicated Azure subscription for better isolation and management. Apply your configurations based on how the subscription is used. If Sentinel is the only workload, configure it at the subscription level. If the subscription hosts multiple workloads, configure it at the resource group level.

Keep in mind that Azure policies, **data collection rules** (**DCRs**), and Logic Apps may exist in separate resource groups. Your home tenant might include roles such as SIEM engineers or automation specialists who need access across different scopes and tenants. Ensure these roles are assigned the necessary permissions accordingly.

Step 3: Assigning active roles

Now we will assign active roles into the groups that users will be able to PIM into, thus giving them the role in a time-boxed, governed way.

1. For each role, navigate to the role in PIM: within Entra ID, navigate to **Privileged Identity Management | Azure resources**. Click the **Manage resource** button, with the relevant subscription set.

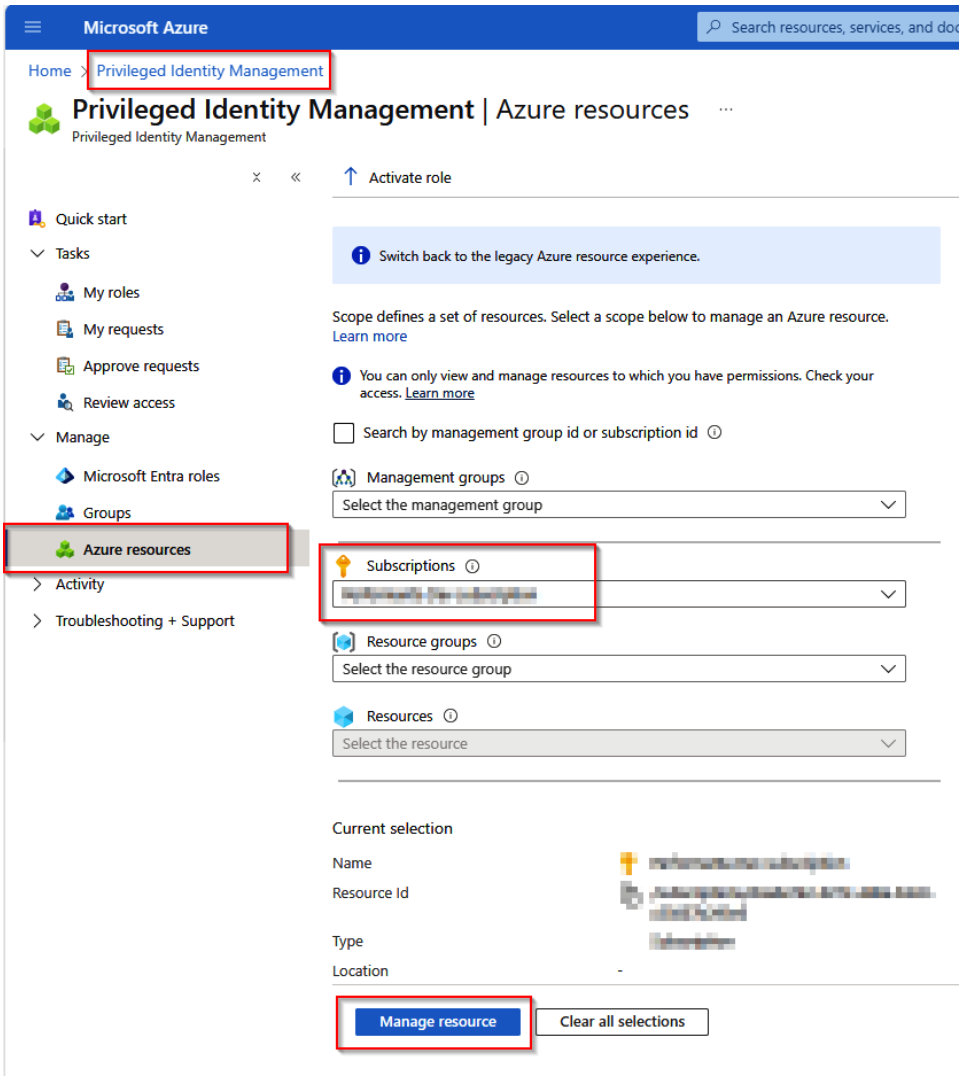


Figure 7.32: Navigate to the role in PIM

2. Add one of the roles mentioned at the top of *Step 2* as an **active assignment** for the corresponding group, for example, **Sentinel Reader – Tier 1 Reader**.

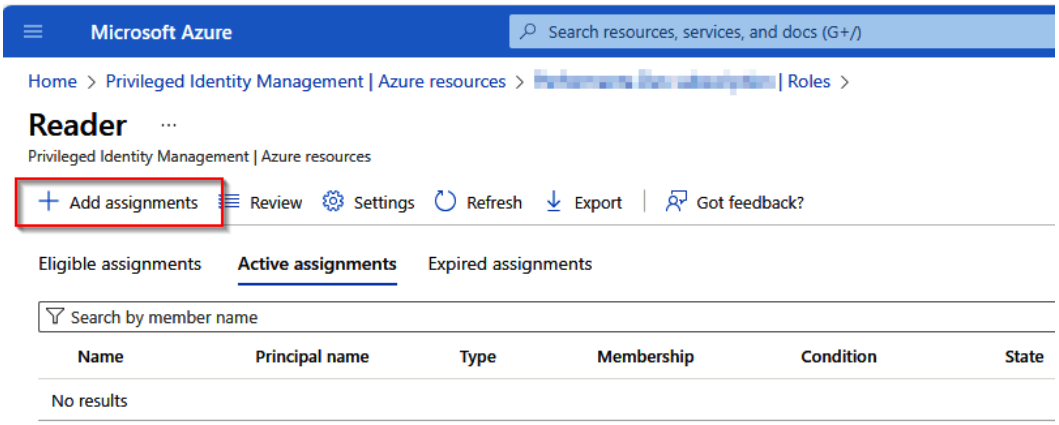


Figure 7.33: Add assignments

Figures 7.34 and *7.35* show how to assign one of the relevant roles to a corresponding group.

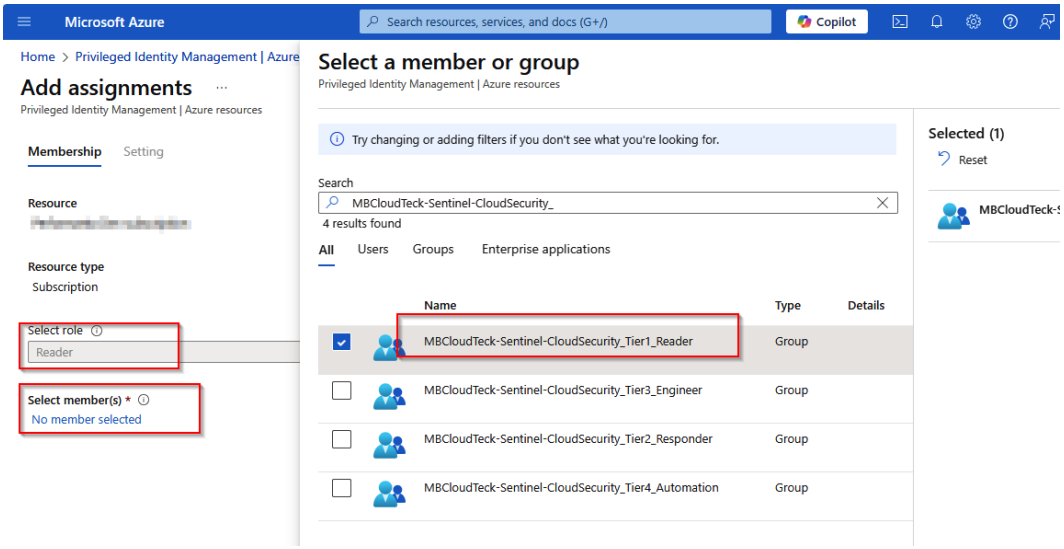


Figure 7.34: Select members

When assigning the roles, make them active and permanently assigned, due to PIM being required to get into the group with this active role assignment. Typically, the role will always be active, but no users will ever be permanently assigned.

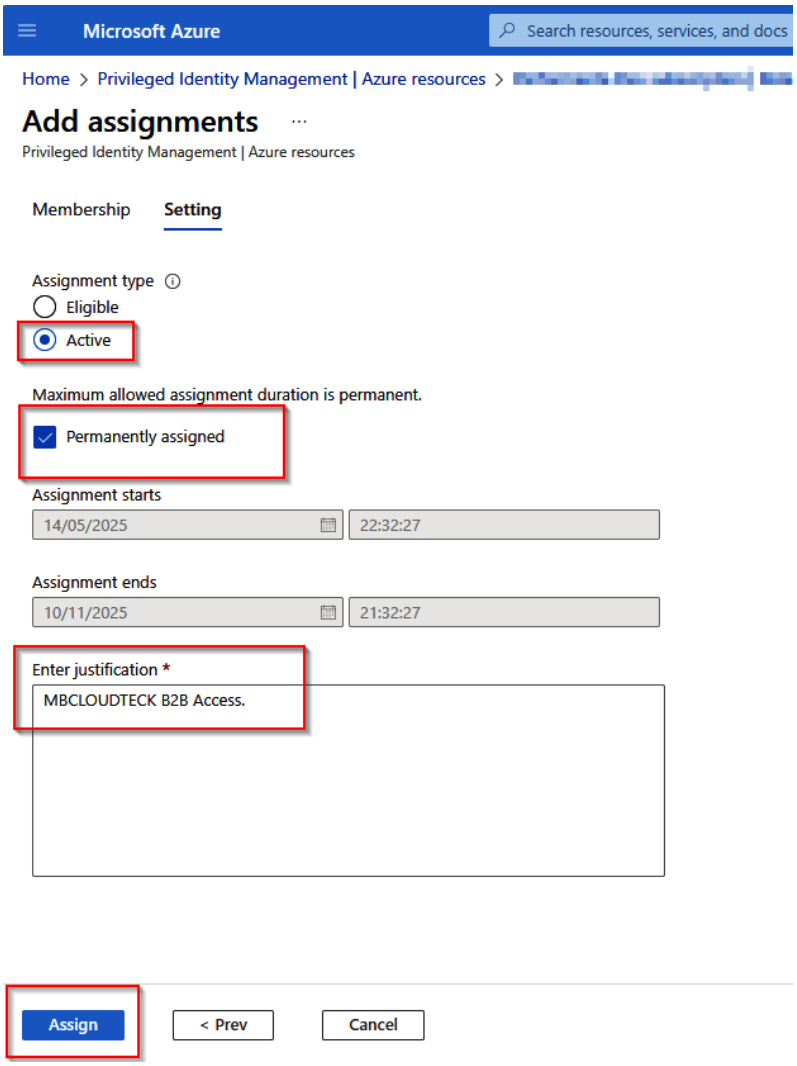


Figure 7.35: Assignment settings

Repeat this process for all applicable Sentinel and Azure roles. At this stage, we recommend that you map your role to groups for clarity in your scenario.

Now, we will look at this same process on the Microsoft 365 side with URBAC roles.

Step 7: URBAC Defender roles

We'll now configure the necessary roles for Defender XDR using URBAC. For the purposes of this demonstration, we'll assume that the following prerequisites are already in place:

- The **Defender XDR Unified platform** has been enabled (we'll cover this in more detail later in the book)
- **URBAC** has been activated across all relevant workloads
- Any existing **legacy roles** have been reviewed and deemed insufficient for current access requirements



If you haven't yet activated **all supported URBAC workloads**, you may assign **Entra ID global roles** directly to the security groups as a temporary measure. This can serve as an interim solution during migration, as seen in the earlier in-house example of setting up security groups in *Chapter 6* where we explained how to delegate if you are only accessing your own Sentinel instance.

Now, let's proceed to create three **custom roles**. With URBAC, there are virtually unlimited configuration possibilities, allowing you to tailor roles to your organization's exact access and compliance needs. These roles are called "custom" because you can precisely choose what actions each role can perform and which Defender products they apply to. This is a meaningful shift from the past, where creating roles with this level of detail often requires command-line configuration.

1. Add a custom role name following a clear naming convention as these names will be seen across Defender XDR and will help with troubleshooting.

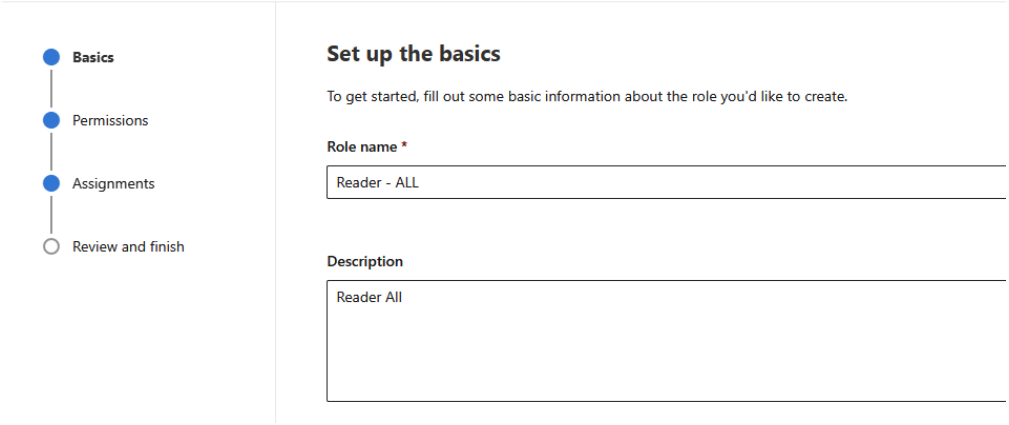


Figure 7.36: Creation of custom roles

2. As you can see, for example, we choose **All read-only selected** access, as we are creating a reader role.

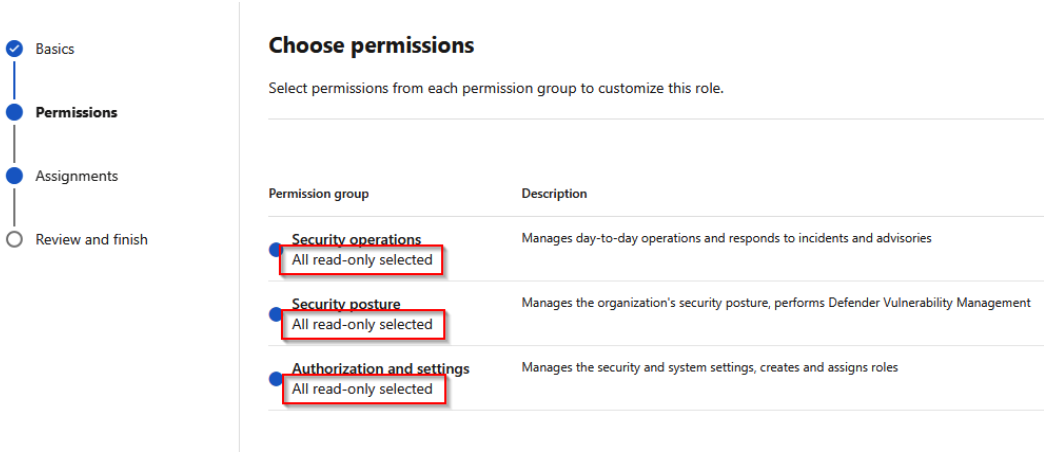


Figure 7.37: Choose permissions 1

3. Fill in the details

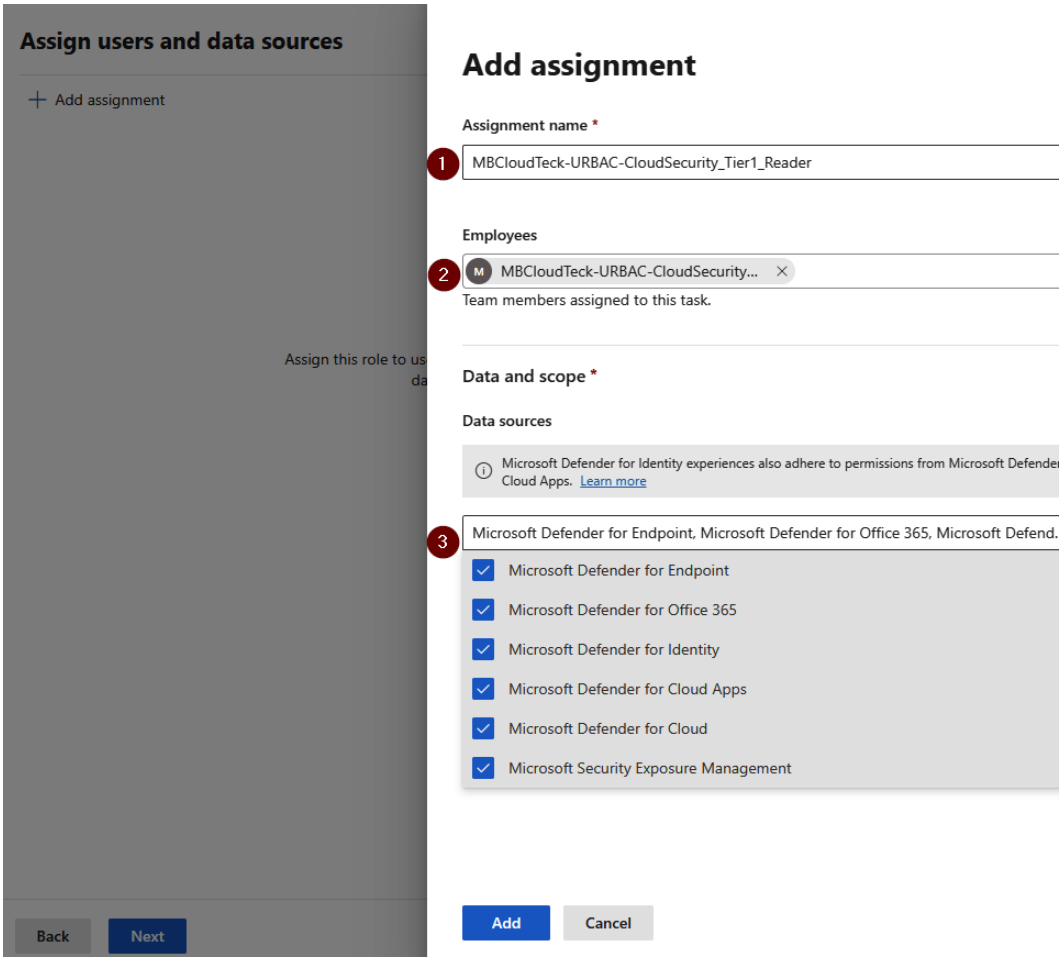


Figure 7.38: Choose permissions 2

4. Click submit to create the role.

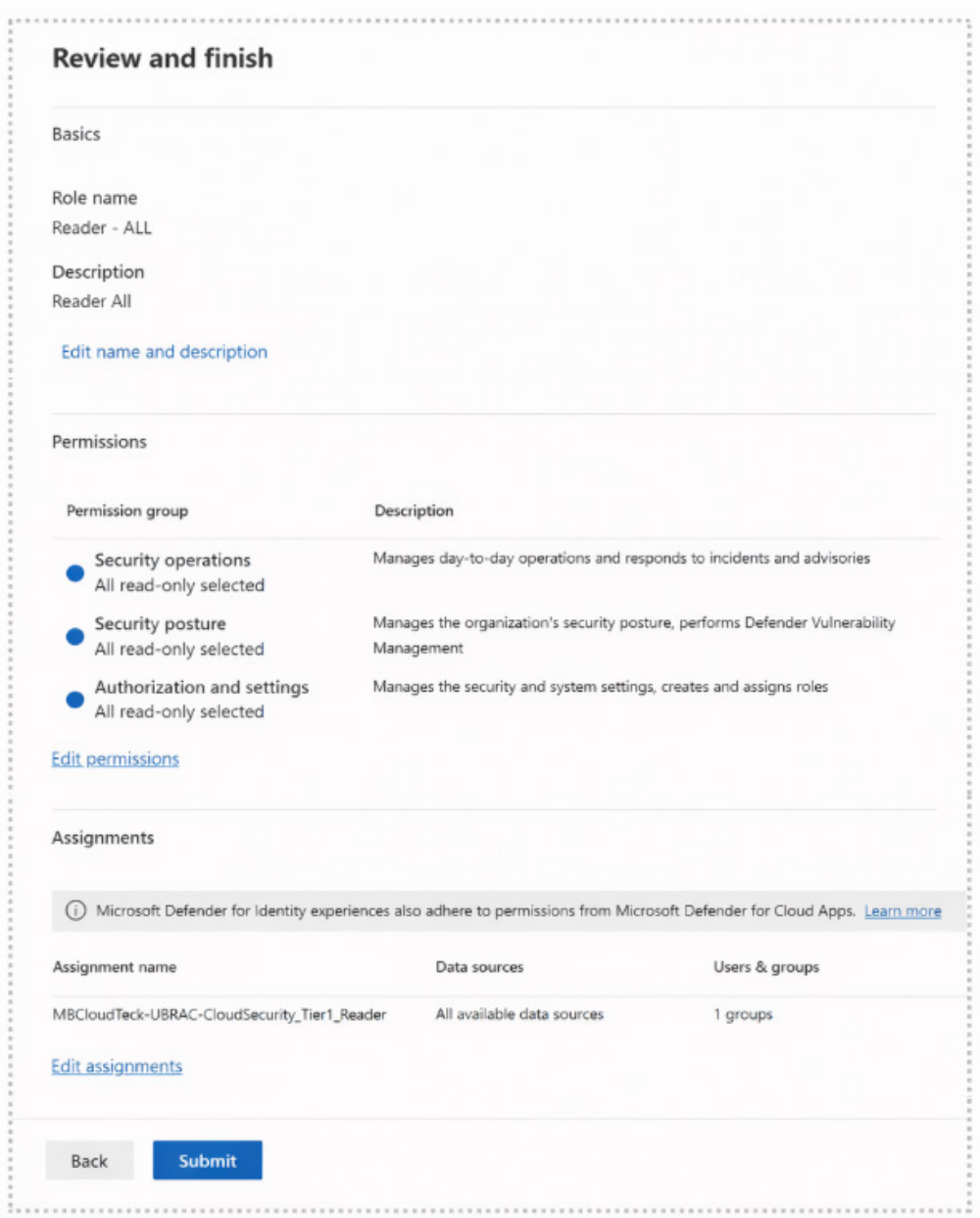


Figure 7.39: Review settings and submit

Repeat this process for each required role.

At this stage, we've provisioned the necessary SGs for both Azure RBAC and Defender XDR URBAC and assigned the relevant permissions to each group. Now, we'll move on to the next step to integrate these groups with APs, so that users from the home tenant can request access via entitlement management.

Step 8: External access governance setup

To allow seamless onboarding of B2B contacts, we must take full advantage of the identity governance feature around AP access and management. Let's create a catalog and make it available to the right users:

1. Navigate to **Microsoft Entra ID | Identity Governance** and select **New catalog**.

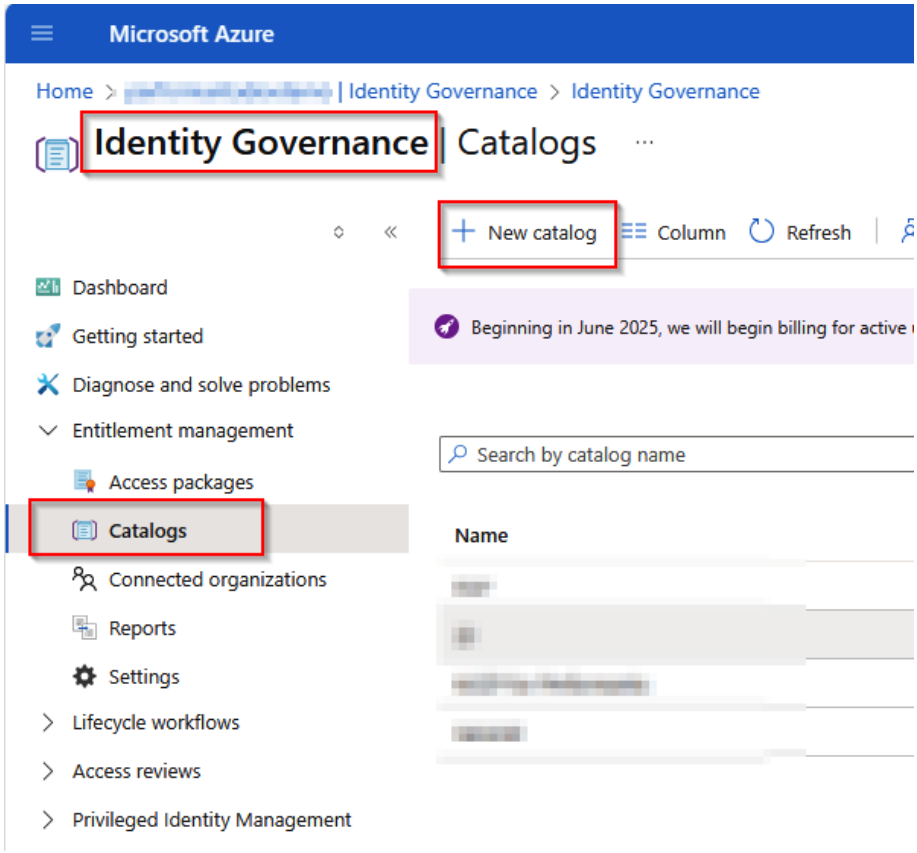


Figure 7.40: Create a new catalog

2. Add the name and description of the catalog in their respective fields. Then enable external user access for the catalog.

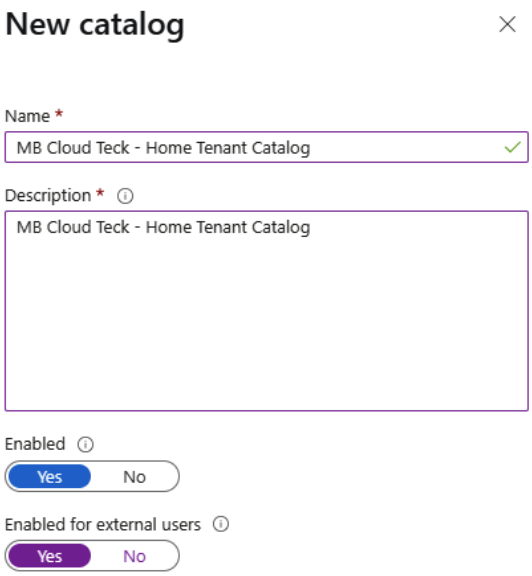


Figure 7.41: Catalog settings

3. Within the catalog, go to **Manage | Resources | Add Resources**. The following figure shows the addition of the previously created AP SGs.

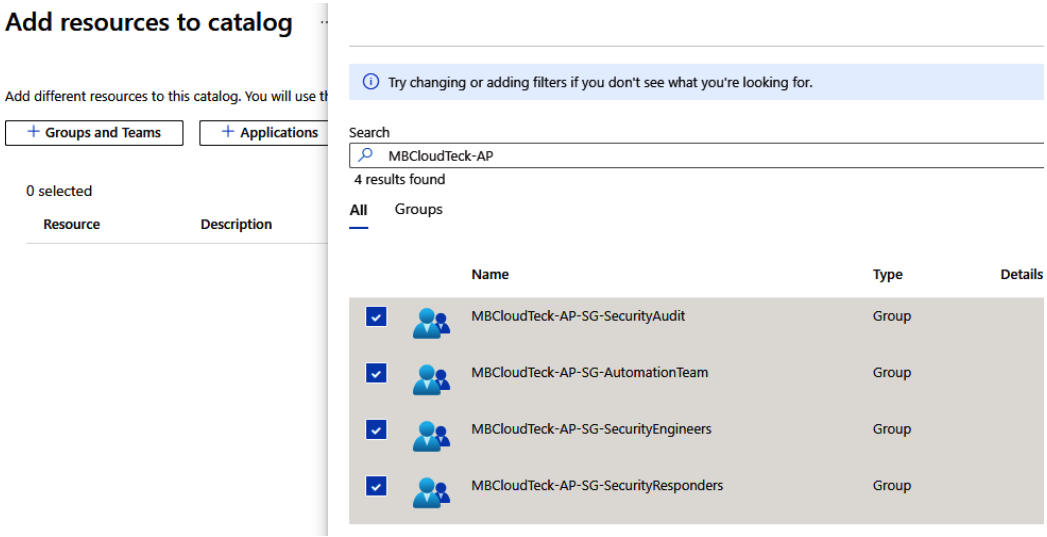


Figure 7.42: Adding resources to the catalog

4. Select the AP SGs created earlier.

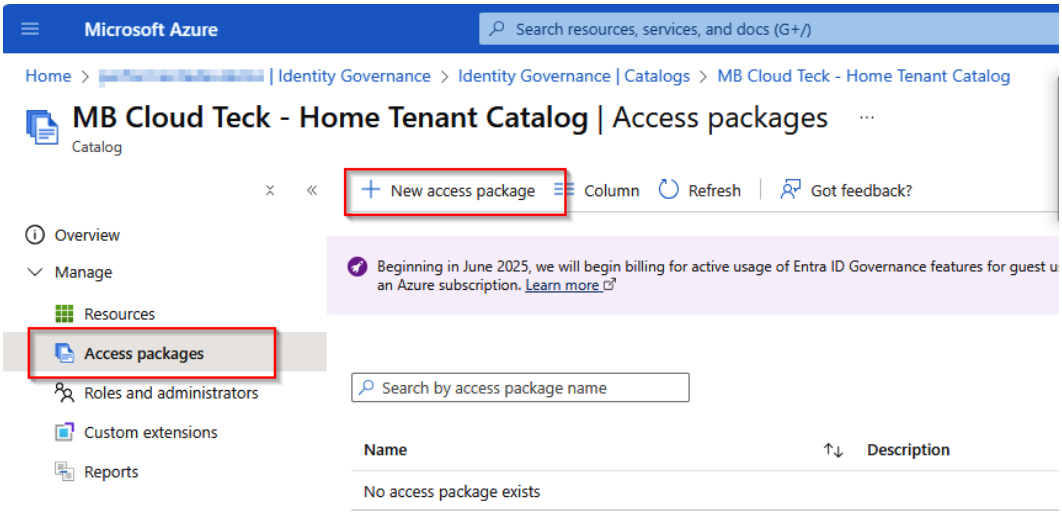


Figure 7.43: Creation of APs per role

5. Create a new AP for each role.

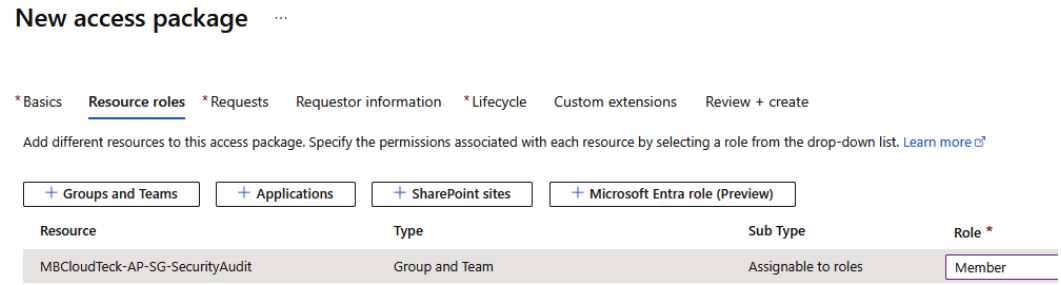


Figure 7.44: Role created

6. When prompted, select the external organization added previously. Add approvers and make sure to include both the external sponsors and internal approvers identified earlier.

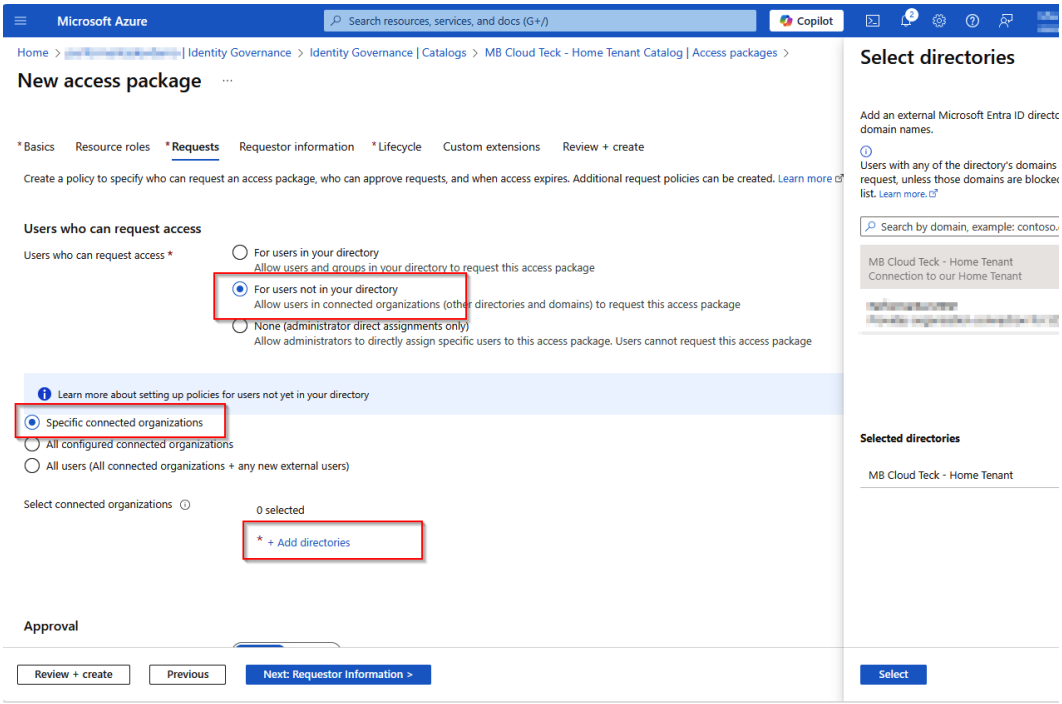


Figure 7.45: New AP settings

7. Configure the lifecycle by enabling access reviews.

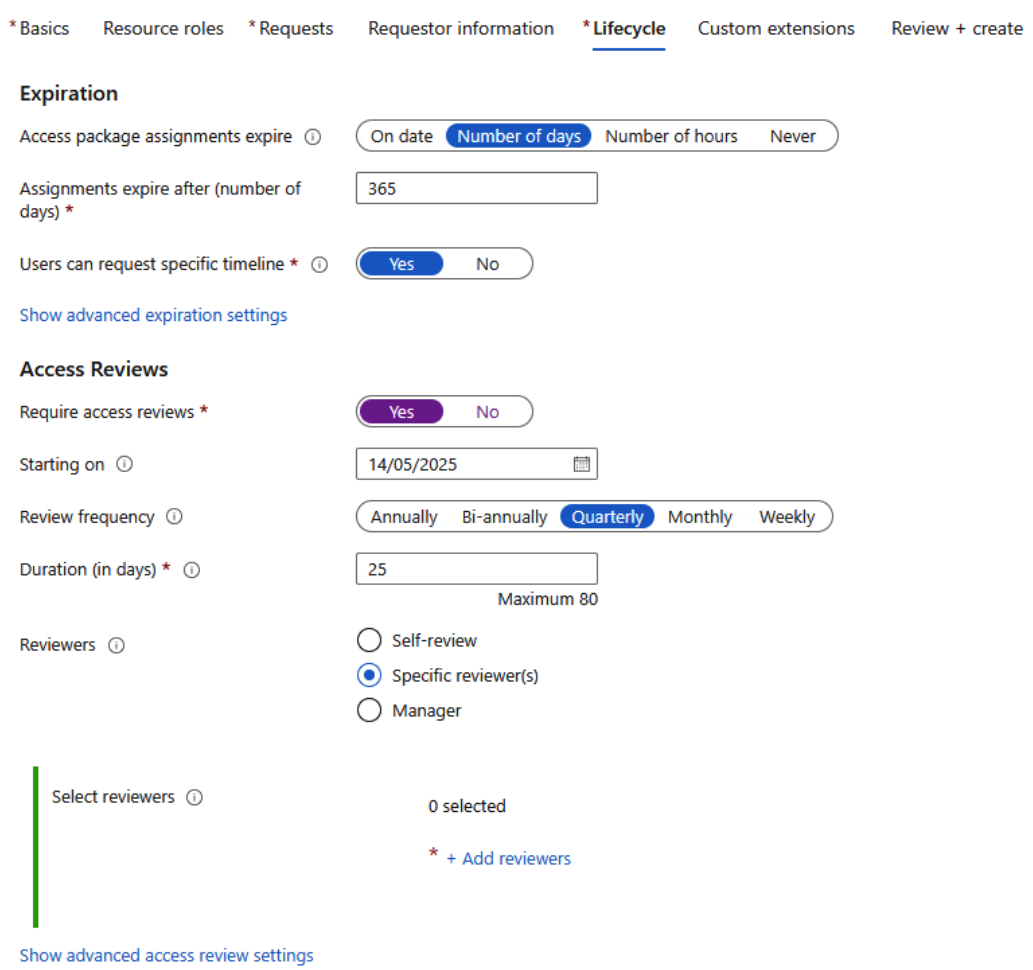


Figure 7.46: AP lifecycle

8. Review and create the package.

You've now successfully created a catalog, defined APs, and enabled external B2B governance with automated reviews and lifecycle management. This setup enables B2B users to seamlessly request and gain access to the client environment securely and with proper oversight. Let's now move on to the final step.

Step 9: Conditional Access authentication context

This step is to enforce passwordless authentication for home tenant PIM elevation. To improve the security posture of your home tenant, we'll now create a Conditional Access policy that enforces passwordless or phishing-resistant authentication during PIM elevation.

Follow these steps to create the policy:

1. Navigate to **Microsoft Entra ID | Security | Conditional Access | Create a new policy**.

   Include all SGs previously created that grant access to APs.

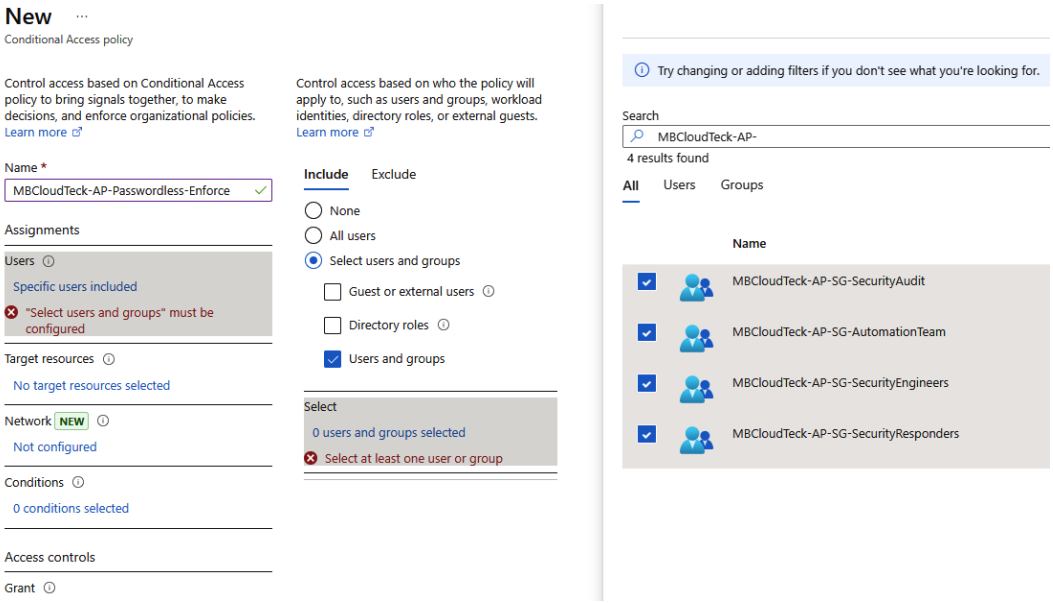


Figure 7.47: Conditional Access policy settings for APs

2. Under **Target Resources**, select the custom authentication context created earlier.

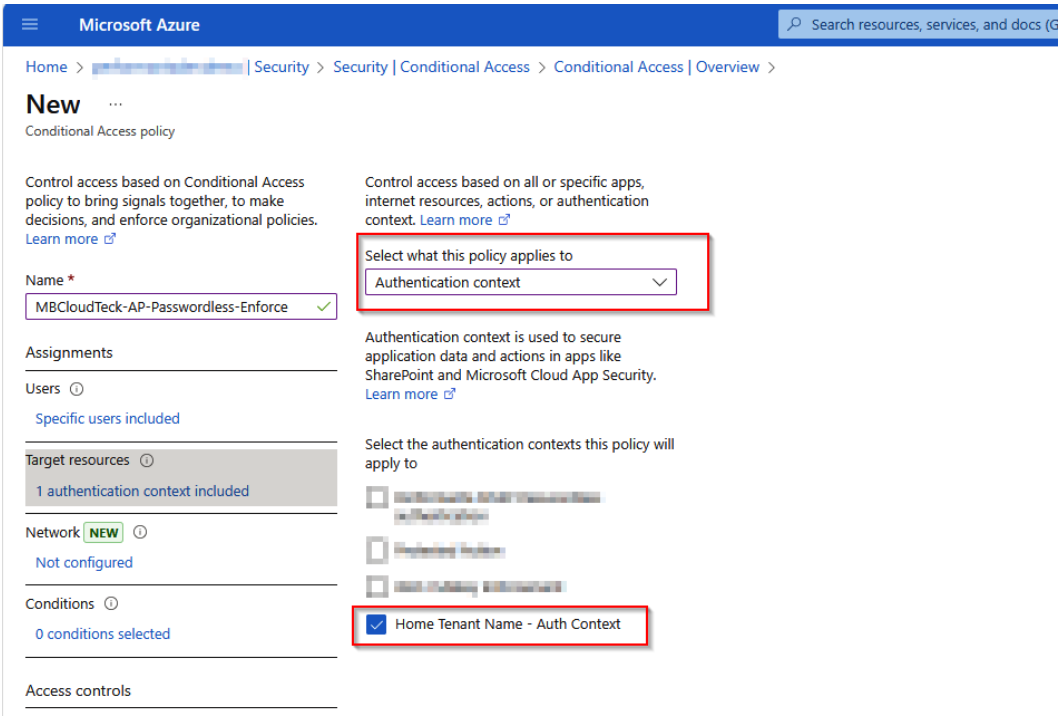


Figure 7.48: Conditional Access policy configuration

3. Select the authentication strength. Choose the **Passwordless** or **Phishing-resistant** option.

4. Set the policy to **On** and then create the policy.

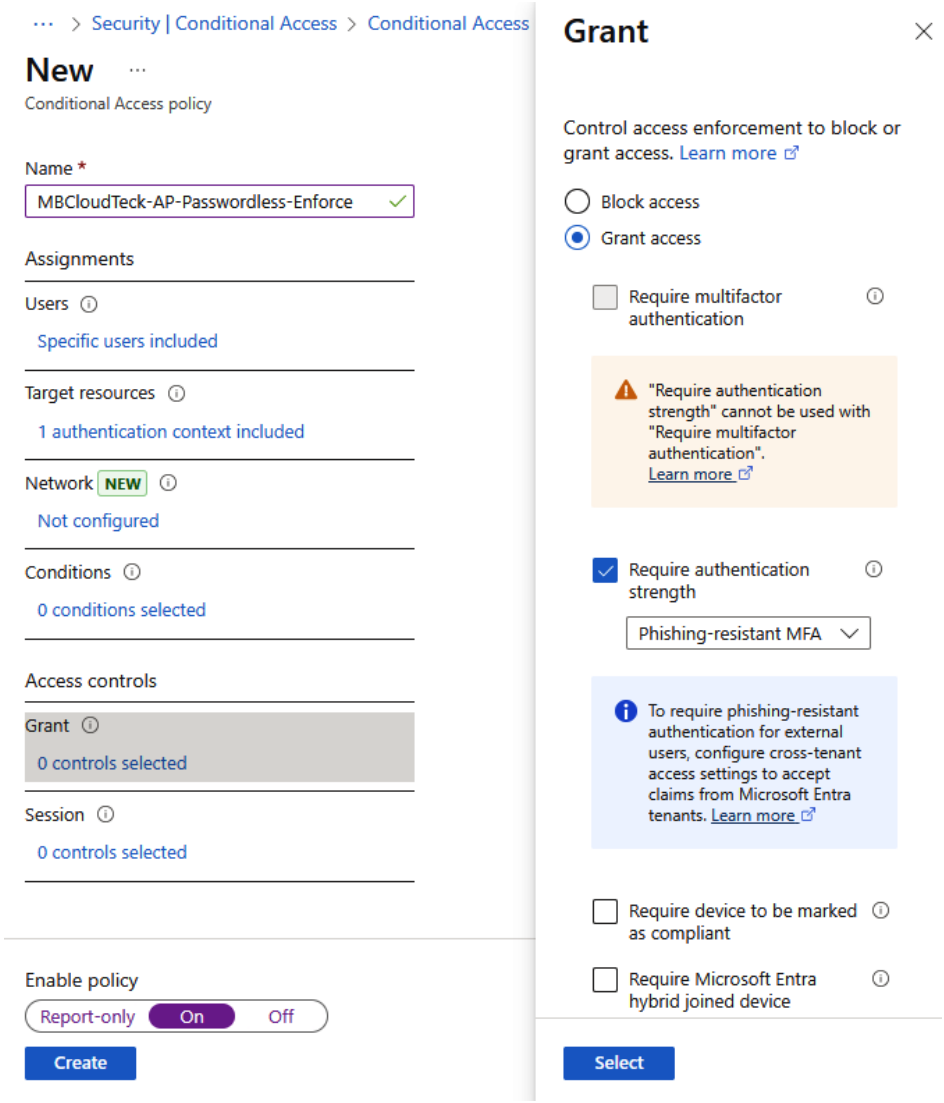


Figure 7.49: Conditional Access policy configuration extended settings

Outcome: Bringing external access experience together

Once everything is configured, here's what the **external (home tenant) user experience** will look like.

The guest user you are inviting receives a link to the AP (typically via email or documentation).

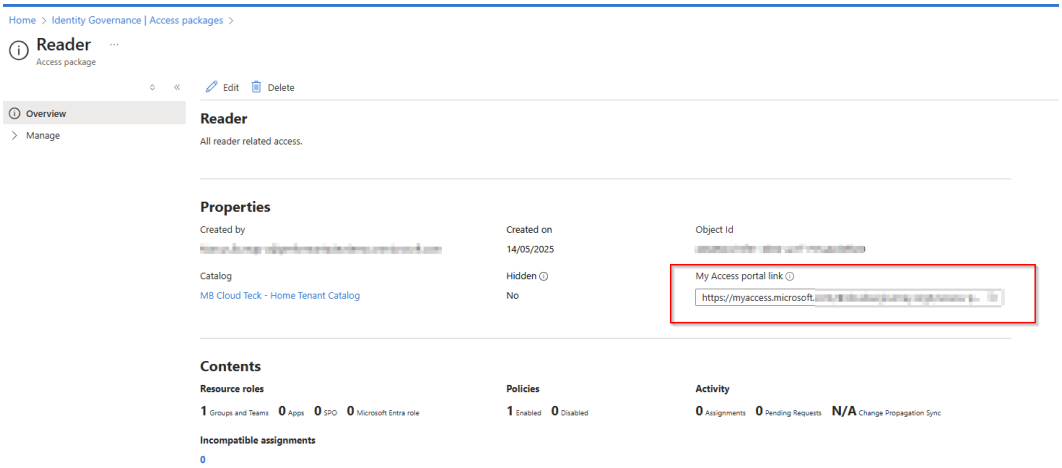


Figure 7.50: AP invitation link

Clicking the link prompts the user to sign in and accept any client tenant terms of use, usage policies, or data processing requirements.

The user can request access via the **My Access** portal and by selecting the available AP. If sponsor approval is required and has not been granted, the package will not appear.

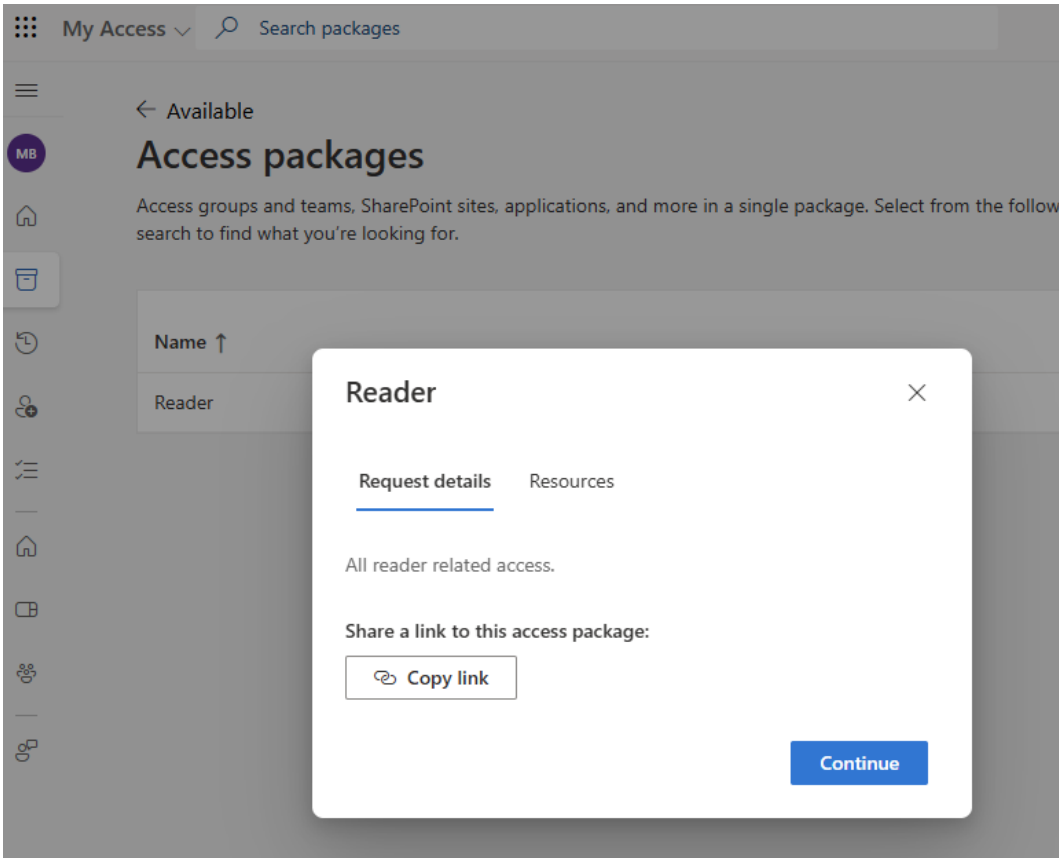


Figure 7.51: My Access portal

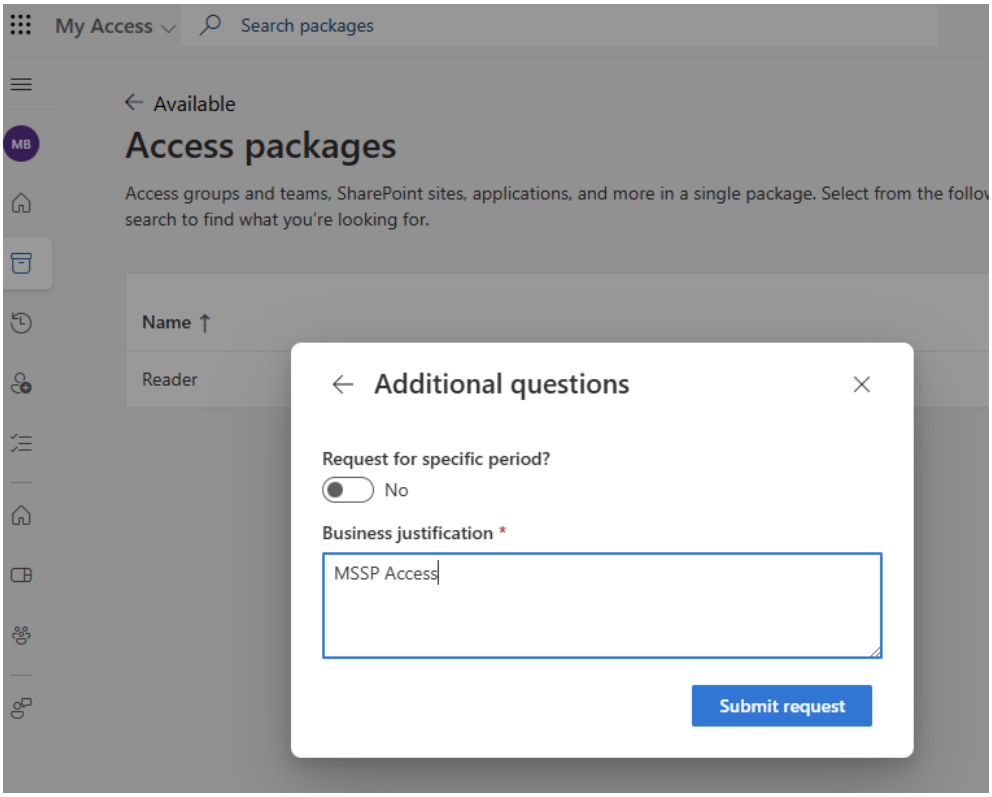


Figure 7.52: AP access request with justification

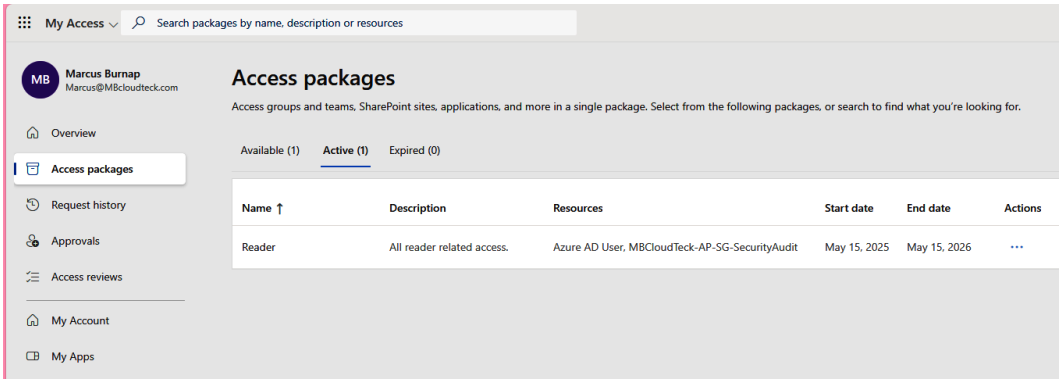


Figure 7.53: Active package on user account

Once access is approved, the user switches to the client directory by selecting their profile icon and navigating to the appropriate tenant.

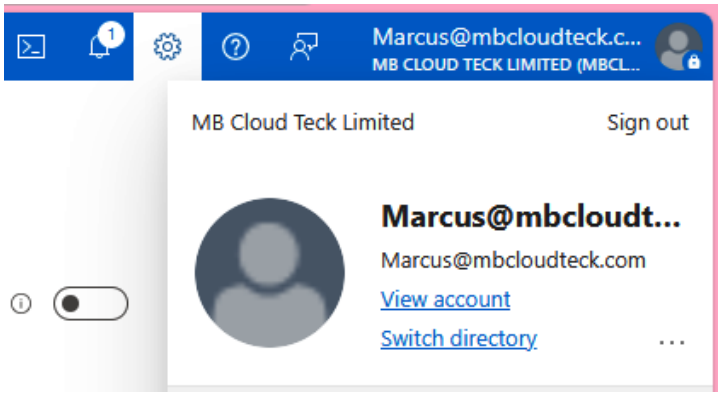


Figure 7.54: Switch directory

Within the client context, the user navigates to PIM. The assigned role should now be visible and ready for activation. Activate the role. It may take a few minutes to become active.

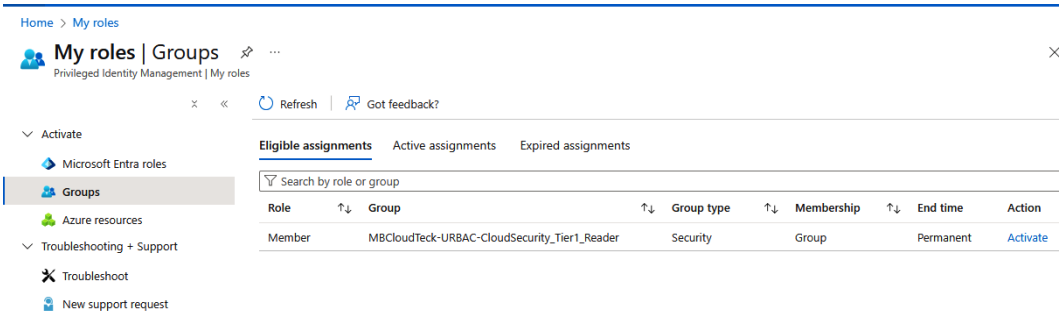


Figure 7.55: Activate PIM

To validate access, navigate to the relevant resource. In this example, the target is a URBAC group, so you would use the Defender portal. Make sure you have switched to the client tenant context.

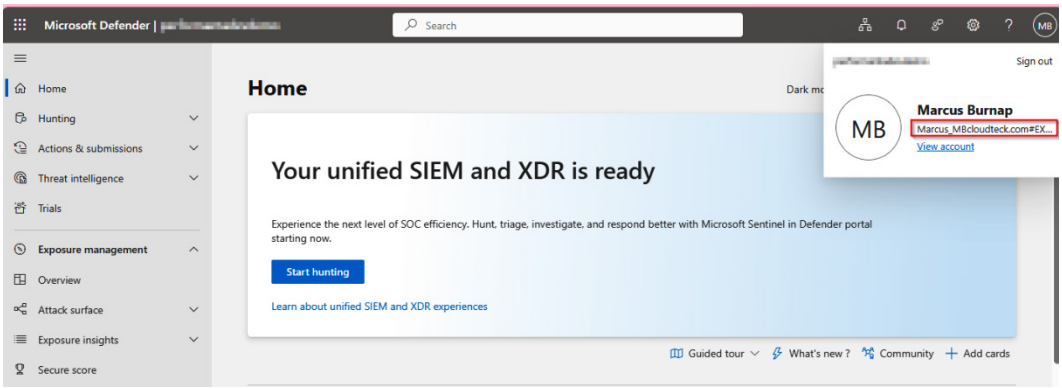


Figure 7.56: Context check

You should now see that unified access is live and functioning correctly for the external user. The same process applies if the user has been elevated into a Sentinel or Azure-related group.

Overall, we have seen how to bring the full experience together from end to end: users request access, switch to the customer directory, activate roles via PIM with the right authentication context and Conditional Access, and then validate access in both Azure/Sentinel and Defender XDR URBAC.

Summary

In this chapter, we examined how MSSPs and large enterprises onboard customers using secure, scalable, and automation-friendly access models. It introduced Azure Lighthouse as the foundation for cross-tenant delegated access, enabling organizations to manage customer environments centrally while maintaining strong governance and separation of duties. The chapter took you through best practices for onboarding using ARM templates, assigning SOC tier roles, and managing entitlements with Entra ID SGs.

It also compared GDAP and the B2B collaboration model, helping you understand when and how to use each access mechanism. These models supported Zero Trust alignment, operational automation, and long-term visibility across customer environments. With delegated access successfully configured, the next logical step is to operationalize Microsoft Sentinel. In the next chapter, we will map out a 90-day deployment journey, guiding you through every stage from telemetry onboarding to advanced threat detection and automation.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



8

Joining the Dots: Microsoft Sentinel Deployment Plan and Features

This chapter lays out a practical Microsoft Sentinel deployment plan, structured as a 30-, 60-, and 90-day delivery cycle. It is intended for the point in the journey where the platform foundation is already in place, the customer is onboarded, and you are ready to turn Sentinel from "installed" into "operational."

A Sentinel deployment plan is not just a project timeline. It is the working blueprint that moves you from initial log onboarding to a functioning detection and response capability. It covers the decisions and builds steps that form the backbone of Sentinel: what data you ingest, how it is normalized and retained, how incidents are created and triaged, how automation is safely introduced, and how hunting becomes a repeatable practice rather than an ad-hoc activity.

This chapter will help your team develop hands-on familiarity with Sentinel's features while establishing a measurable security impact. By the end of this chapter, you will know how to formulate a phased plan that accelerates time-to-value while building a SOC-ready capability that can be governed, measured, and improved over time.

In this chapter, you'll learn about the following key topics:

- Permissions in the Unified Microsoft Defender Portal
- Microsoft Sentinel SIEM deployment plan
- Executing your deployment plan for Microsoft Sentinel
- The backbone of Microsoft Sentinel: Introductions to KQL, detection rules, entity mapping, summary rules, automation, incidents, workbooks, hunting, threat intelligence, and the MITRE ATT&CK framework

Let's begin with the permissions in the Unified Microsoft Defender Portal

Permissions in the Unified Microsoft Defender Portal

In July 2025, Microsoft announced that the Microsoft Sentinel experience will be retired in the Azure portal on July 1, 2026. With the introduction of the unified Microsoft Defender portal experience, one of the most common areas of concern is permissions. Customers consistently ask the same questions:

- How are permissions managed going forward?
- What happens to my existing Azure permissions?
- Do I still need Microsoft Sentinel specific roles such as Sentinel Contributor?

This section clarifies how permissions work across Microsoft Sentinel and Microsoft Defender XDR in the unified portal, what stays the same, what changes, and how SOC teams should plan access going forward.

Microsoft Sentinel SIEM permissions remain unchanged

Microsoft Sentinel continues to rely on **Azure RBAC** for access control. All existing Sentinel roles and permissions remain exactly as they were in the Azure portal. If a user can access Sentinel data, incidents, or logs in the Azure portal, they will continue to see the same data in the unified Defender portal.

Common Sentinel roles include **Microsoft Sentinel Reader** for reading Sentinel data, **Microsoft Sentinel Responder** for incident response and automation rules, and **Microsoft Sentinel Playbook Operator** for manually executing playbooks. No permission changes are required purely because of the new portal experience. Sentinel permissions are still assigned at the workspace level through Azure.



There are no new Sentinel roles and no replacement for Sentinel Contributor. Existing Sentinel roles remain valid and required.

Microsoft Defender XDR permissions are separate

For customers who previously worked only in the Sentinel Azure portal and did not use the Defender XDR portal, this is where behavior may feel different. Access to **Microsoft Defender XDR** features such as alerts from Defender for Endpoint, Office, Identity, or Cloud; XDR tables in advanced hunting, and native Defender portal experiences requires Defender specific permissions. These permissions are not granted by Sentinel roles.

You can grant access in two ways:

- Broad Microsoft Entra ID built in roles such as Security Reader or Security Administrator
- Unified RBAC (URBAC) roles scoped only to the Defender portal

URBAC is strongly recommended because it limits access strictly to security workloads and avoids over permissioning SOC users across Microsoft 365 and Azure.

Unified RBAC currently supports Microsoft Defender XDR, exposure management, Defender for Cloud in the defender portal, Security Copilot investigation and response, and Microsoft Sentinel data lake using the default workspace. Support for Microsoft Sentinel SIEM actions through URBAC is coming but is currently not available. This means Sentinel SIEM permissions still require Azure RBAC today.

Minimum URBAC role for Defender XDR investigations

To allow you to view and manage Defender XDR incidents, alerts, and query all Advanced Hunting tables, create a custom URBAC role with security data basics read and alerts management. This role is created in the Defender portal under: **System | Permissions | Microsoft Defender XDR | Roles**.

Special workload considerations

The following are some of the special workload considerations you need to keep in mind:

- **Defender for Endpoint device groups**: If device groups are configured, access must be explicitly granted to analyst Entra ID groups under Endpoint roles and groups. Without this, devices and alerts will be hidden.

- **Defender for Identity scoping**: If access needs to be limited to specific domains or organisational units, Identity scoping is applied during role assignments. This is critical for large or segmented Active Directory environments.

Consider the following common SOC access patterns as well:

- **Sentinel only analyst**: If an analyst only needs Sentinel detections, logs, workbooks, automations, and content hub access:
    - Assign Microsoft Sentinel roles in Azure
    - Optionally assign Sentinel Playbook Operator

    They will not see any Defender XDR alerts or incidents.
- **Defender XDR only analyst**: If an analyst only needs Defender incidents, alerts, and XDR hunting: assign a URBAC role or Entra ID security role. They will not see any Sentinel data.
- **Unified SOC analyst**: If an analyst needs access to both Sentinel and Defender XDR:
    - Assign Sentinel roles in Azure
    - Assign Defender XDR permissions using URBAC

Unlike the Azure portal, a Sentinel role alone is no longer sufficient for full visibility.

- **Sentinel only customers without Defender XDR licenses**: After initial onboarding using Security Administrator or Global Administrator:
    - No additional permissions are required to continue using Sentinel
    - Defender native features such as Threat Analytics and Case Management will not be available

When an incident contains alerts from multiple products and the analyst lacks permission for some of them, only permitted alerts will be visible, a banner indicates limited visibility, and incident management actions are disabled. You can still work on individual alerts they are authorised to see, but only users with full visibility can manage the incident as a whole.

Managing multiple Sentinel workspaces

Workspace level permissions continue to apply. If a user is granted access to only one Sentinel workspace, they will only see data from that workspace, even if multiple workspaces are connected to the unified portal.

DLP and Purview considerations: DLP incidents in the Defender portal behave differently depending on permission model. Global Entra ID roles provide visibility automatically and URBAC users require Purview RBAC roles. Purview RBAC is federated into the Defender portal but is not managed by URBAC itself.

Migrating away from global Entra ID roles: Microsoft provides mappings to recreate global security roles using URBAC. The following is the recommended approach:

1. Create equivalent URBAC roles
2. Enable Defender workloads in URBAC
3. Remove global Entra ID roles from users

Some actions such as connecting Sentinel workspaces, changing primary workspaces, onboarding first party connectors, and accessing the Sentinel data lake still require global roles. URBAC coverage continues to expand, but global roles are not fully eliminated yet.

Legacy RBAC and rollback safety

Standalone RBAC for individual Defender products remains supported. Customers are encouraged to migrate gradually. If a URBAC configuration error blocks access, URBAC can be toggled off and standalone RBAC immediately resumes control. This provides a safe rollback during transition.

Privileged Identity Management and auditing

URBAC supports PIM for groups, not individual roles. Users activate membership in a privileged group to gain Defender access. URBAC activity is audited through Defender XDR audit logs, Purview audit logs, and CloudAppEvents table in Advanced Hunting. This provides full visibility into role creation, modification, and deletion.

The unified Defender portal does not simplify permissions by collapsing them into a single model. Instead, it makes the separation explicit:

- Microsoft Sentinel SIEM access remains Azure RBAC based
- Microsoft Defender XDR access is governed by URBAC or Entra ID roles
- Full SOC visibility requires both models

Designing access intentionally is now a core part of operating Sentinel and Defender at scale.

Now that we have touched based on the new set of permissions needed for managing Sentinel in the Unified Microsoft Defender Portal, let's look at the deployment plan.

Microsoft Sentinel SIEM deployment plan

The first step in any Microsoft Sentinel deployment is to get the basics right. That means setting up the telemetry pipeline, connecting key data sources, and making sure that the foundation is solid. Without this, everything else falls short. This is where your journey starts.

30-60-90 day deployment plan: from setup to security outcomes

The first 90 days of deployment are about getting Sentinel up and running in a way that's practical and sustainable. Each stage builds on the last:

- **Days 0–30**: Core setup and initial data collection
- **Days 31–60**: Tuning data, building detection rules, and adding automation
- **Days 61–90**: Enabling advanced features such as threat hunting, ML-based detection, and cross-platform analytics

This phased approach keeps your team focused, helps build confidence, and delivers real outcomes early on. Let's dive into the plan!

Days 0–30: Build the foundation – data collection and connector configuration

Table 8.1 shows some of the key actions you must take in this phase and the purpose of those actions:

Focus area	Actions	Purpose
Connect core Microsoft Services	Connect Microsoft 365, Defender XDR, Azure activity logs, and Entra ID. Prioritize these for quick, high-value coverage.	Provide immediate visibility into identity, endpoint, and cloud activity.
Extend to third-party and custom sources	Use Content Hub for firewalls, proxies, and SaaS. Use CCP for unsupported tools via schema-based HTTP posts (no code).	Cover hybrid, legacy, or edge systems without complexity.

Focus area	Actions	Purpose
Validate data ingestion	Use KQL to check for fresh timestamps, expected fields and values, and normal log volume patterns.	Catch ingestion issues early to avoid detection gaps.
Enable analytics rules	Turn on Microsoft-curated rules mapped to MITRE ATT&CK. Start with defaults and tune thresholds as needed.	Begin detecting common threats immediately.
Use Workbooks for initial visibility	Leverage built-in dashboards for connector health, data flow, and incidents.	Give SOC teams fast visual feedback without building custom visuals.
Integrate with Microsoft Defender XDR	Connect Defender XDR to unify incidents and enable cross-platform correlation and hunting.	Speed up investigations and improve response with richer context.
Automate initial response workflows	Create and test Logic App playbooks. Define automation rules for triggering based on incident details. Manually test playbooks before enabling auto-execution.	Establish repeatable, scalable response actions early.
Recommend starter playbooks	**Notification**: Alert via Teams, email, or tickets. **Blocking**: Disable accounts, block IPs, and isolate hosts. **Incident management**: Create/update/close incidents in external systems.	Cover high-impact actions early with proven response patterns.

Table 8.1: Day 1–30 actions

Days 31–45: Optimize collection and enhance visibility

With the core pipeline in place, the next step is optimization. This means reducing noise, improving cost-efficiency, and expanding telemetry to close any gaps in coverage. *Table 8.2* shows some of the key actions you must take in this phase and the purpose of those actions:

Focus area	Actions	Purpose
Leverage SOC optimization insights	• Identify high-cost, low-value data sources. Measure each source's contribution to detections and investigations. • Prioritize data based on real threat exposure. • Use the SOC Optimization API across systems.	Improve efficiency by focusing on data that delivers real security value.
Integrate missing or underused data sources	• Reconnect or adjust skipped log sources. • Add firewalls, DNS logs, or EDR telemetry. • Ensure that connectors are healthy and logs are usable.	Achieve full-spectrum visibility across identity, endpoint, cloud, and network.
Refine ingestion with data collection rules (DCRs)	• Filter out unnecessary logs. Tag records with metadata (e.g., region and environment). • Mask sensitive fields for compliance. Route data to custom tables. • Use Syslog or CEF for Linux and custom systems.	Make ingestion efficient, compliant, and ready for scale.

Table 8.2: Day 31–45 actions

Days 46–60: Deepen detection, investigation, and response

Now that Sentinel is collecting meaningful data, it's time to go deeper into custom detections, rich investigations, and smarter automation. *Table 8.3* shows some of the key actions you must take in this phase and the purpose of those actions:

Focus area	Actions	Purpose
Build custom analytics and enable fusion	• Create custom analytics rules for insider threats, industry risks, or known behaviors • Use rule templates for quick setup • Enable Fusion for cross-domain correlation using ML	Detect complex attacks spanning identity, endpoint, and cloud systems
Develop advanced workbooks	• Customize templates for detection trends and KPIs. • Build dashboards for SOC, execs, and investigations • Add real-time telemetry for active monitoring	Turn raw data into usable insights for analysts and leaders
Enrich incidents and correlate signals	• Group related alerts into one incident. • Correlate across Defender, M365, and custom sources • Use playbooks to enrich incidents	Reduce alert noise and give analysts more context faster
Strengthen threat hunting	• Run regular (daily/weekly/monthly) hunting queries • Use past insights to guide new hunts • Align with MITRE ATT&CK	Proactively uncover subtle or emerging threats
Expand automation strategy	• Triage incidents by severity, asset, or source • Use Logic Apps for response automation and reduce manual workload	Scale response while improving consistency and Mean Time to Respond (MTTR)
Pilot Microsoft Security Copilot	• Use natural, everyday language to create queries and summaries • Get context-aware investigation help from Security Copilot • Use AI-generated summaries	Accelerate detection and response with AI-powered workflows

Table 8.3: Day 46–60 actions

Days 61–75: Innovate data collection and enable long-term insights

At this stage, turn your focus to long-term value, retaining data for forensic readiness, streamlining ingestion, and enabling smarter analytics. *Table 8.4* shows the various options around log ingestion.

Focus area	Actions	Purpose
Enable advanced log tiers	• Use the Archived Logs tier to retain logs for up to 12 years for compliance and forensic readiness	Preserve broad data coverage while controlling costs and storage use
Try sentinel data lake	• Ingest noisy telemetry such as DNS or firewall logs using the Auxiliary Logs tier • Use summary rules to condense data into custom log tables	Efficiently handle low-fidelity data and build baselines for anomaly detection
Enhance data transformation workflows	Use Data Collection Rule (DCRs) for the following: • Filter out irrelevant events before ingestion • Enrich logs with tags (e.g., business unit, region) • Normalize fields for consistent querying	Improve data quality and make downstream queries, rules, and dashboards more effective

Table 8.4: Day 61–75 actions

Days 76-85: Strengthen detection with intelligence and analytics

Now, you're ready to integrate threat intelligence, consolidate SecOps tooling, and layer in more advanced analytics capabilities.

Focus area	Actions	Purpose
Integrate cyber threat intelligence (CTI)	• Ingest threat indicators via native connectors • Enhance analytics rules with CTI for contextual alerts • Populate threat intelligence workbooks	Add threat context to detections and improve situational awareness
Enable Unified SecOps with Microsoft Defender	• Connect Sentinel to Defender XDR • Gain unified incident queues, cross-platform disruption, and shared threat hunting • Enable integration via the Defender portal	Streamline investigation and response across Microsoft security tools
Leverage Microsoft Security Copilot	• Use natural language for investigation and reporting • Get context-aware recommendations • Access integrated incident summaries	Use AI to boost analyst speed, accuracy, and decision-making
Use advanced threat detection tools	• Enable User Entity Behavior Analytics (UBEA) to model user and entity behavior • Tune ML-based anomaly detection templates • Explore BYO ML for custom models	Detect insider threats and tailor detection with ML

Table 8.5: Day 75–85 actions F

Days 86–90: Empower analysts with proactive tools

Round out the plan by empowering your SOC analysts with advanced investigation and automation tools.

Focus area	Actions	Purpose
Use Jupyter Notebooks for hunting	• Extend investigations beyond the Sentinel UI • Run Python + KQL queries and visualize data • Use prebuilt notebooks from Microsoft Sentinel GitHub	Model complex attack patterns and perform deep-dive investigations
Automate threat response	• Convert manual playbooks into automated workflows • Trigger automation from analytics rules • Automate actions such as incident triage, IP/user blocking, analyst notifications, or ticketing integration	Scale response actions and reduce time-to-containment

Table 8.6: Day 86–90 actions

Once you have a 90-day plan for your Microsoft Sentinel deployment, you move from "setting up Sentinel" to running a controlled delivery program. The plan becomes your execution blueprint: it tells you what to build first, what to defer, what success looks like at each stage, and which decisions must be made before you switch on detections, automation, and response processes.

With the plan in place, you can start executing it in a measured way. You begin by validating the foundations: confirm that the right data sources are connected, verify data quality and routing, and align retention and cost expectations so the workspace behaves predictably. From there, you can implement detections and incident workflows with confidence, because you are working against a known data baseline and clear operational outcomes. Finally, you introduce automation and hunting as deliberate capability uplifts, not as ad-hoc additions, ensuring that they are safe, repeatable, and auditable.

The key benefit of working from a 90-day plan is control. It prevents random connector sprawl, inconsistent analytics, and rushed playbooks. It also gives you a way to measure progress: you can track improvements in coverage, signal quality, response time, and operational readiness as you complete each milestone.

Executing your deployment plan for Microsoft Sentinel

With your 90-day plan in place, it's time to move from strategy to execution. The next step is deploying Microsoft Sentinel into your environment, which sets the stage for everything that follows. Whether you're starting from scratch or integrating it into an existing Azure setup, this section shows you how to deploy Microsoft Sentinel.

We'll start by covering the essentials. Before deploying Sentinel, make sure to go through the following tasks:

- **Confirm prerequisites**: Active Azure subscription, **Log Analytics Workspace (LAWS)**, and Owner or Contributor + Log Analytics Contributor permissions
- **Define goals and coverage**: Align with business and security objectives and identify key coverage areas such as identity, endpoints, cloud, and applications
- **Select a region**: Choose an Azure region based on compliance, governance, and data residency requirements

If you prefer manual deployment, follow the next steps to set up the core components.

Manual deployment of LAWS and Microsoft Sentinel

Follow these steps to set up the core components:

Step	Description	Purpose
Create or reuse a LAWS	Set up a new LAWS or reuse an existing one. Apply security-focused data retention settings.	Ensure that the workspace is optimized for storing and analyzing security telemetry.
Enable Microsoft Sentinel	Activate Sentinel on the chosen workspace via the Azure portal.	Turn on SIEM and SOAR capabilities in your environment.
Apply role-based access control (RBAC)	Assign RBAC roles such as Reader, Contributor, or Sentinel-specific roles to analysts and engineers.	Provide the right level of access for secure and effective operations.

Step	Description	Purpose
Link to Defender for Cloud	Integrate with Microsoft Defender for Cloud if used. Unlock included data ingestion allowances.	Reduce Sentinel data ingestion costs and enable deeper threat visibility.

Table 8.7: Manual Sentinel Deployment Steps

Creating a LAWS

The LAWS is the foundation of Sentinel. It stores and manages the telemetry your SOC will use for detection, investigation, and response. Follow these steps to create a workspace:

1. Sign in to the Azure portal (`https://portal.azure.com`) using an account with the required administrative permissions
2. In the search bar at the top of the portal, type `Log Analytics Workspaces` and select your primary security LAWS from the results

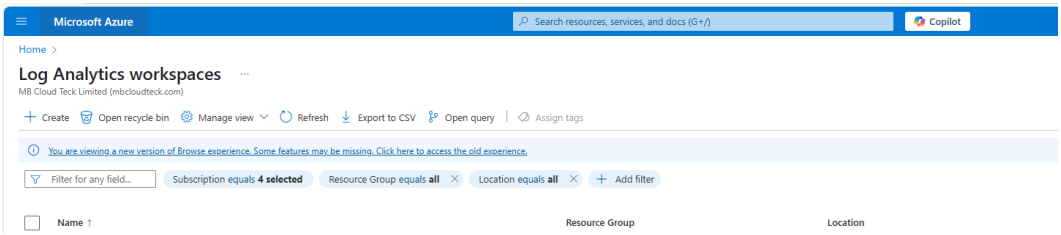


Figure 8.1: Log Analytics workspaces view

3. In the Azure portal, click **+ Create** to start the setup process and initiate the workspace.
4. Now, as shown in *Figure 8.2*, configure the workspace settings by filling out the following fields on the **Basics** tab:
    - **Subscription**: Select the Azure subscription where the workspace will reside
    - **Resource group**: Choose an existing resource group or create a new one (e.g., `rg-security`)
    - **Name**: Enter a meaningful name for your workspace (e.g., `law-sentinel-uksouth`)
    - **Region**: Select the region that aligns with your operations and compliance requirements

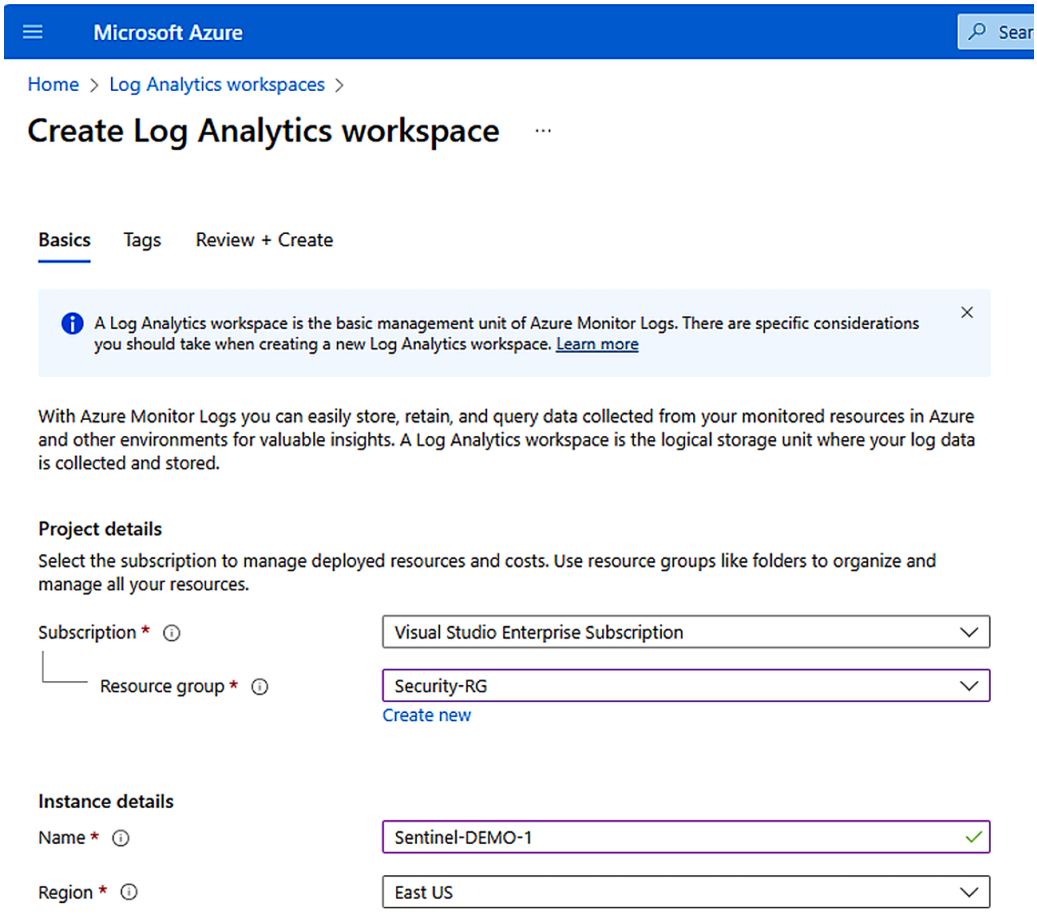


Figure 8.2: Creation of a LAWS



Microsoft Sentinel must be deployed in the same region as your LAWS.

5. Review all the settings, then click **Create** to deploy the workspace.

   Here are some best practices:

   - **Naming convention**: Use a consistent naming scheme (e.g., `law-[purpose]-[region]`) for easier management and identification. It is always good to have the company name in there as well for future lighthouse visibility (when you can see instances from various clients).

- **Tags**: Add governance-related tags during creation, such as `Environment=Security` or `CostCenter=SOC`.

  These tags can prove invaluable for identification and cost management in your Azure subscription.

- **Region alignment**: Always select a region that matches your data residency requirement (typically, the main 365 tenant location). Cross-region Sentinel integration is not supported. This means the LAWS is the location your sentinel exists!

For Microsoft Sentinel, your core residency control is the Azure LAWS region. Raw data is stored in the same region as that workspace, which is why workspace placement is a first-order compliance decision. However, when you onboard Sentinel into the Defender portal (unified SecOps experience), it is important to note that processed data and configuration data might be stored and processed in Microsoft Defender XDR regions.

For Microsoft Defender XDR, the service runs in a specific geographic region for your tenant, and that region is shown in the Defender portal; once created, the Defender XDR tenant can't be moved to a different region.

The practical takeaway, especially when pairing Sentinel and Defender XDR in a unified SecOps model, is simple: do not assume ADR coverage equals security platform residency. Validate Sentinel workspace placement, confirm the Defender XDR region, and understand what shifts (or may shift) when you integrate Sentinel into the Defender portal so your compliance position is deliberate rather than assumed.

Follow this link to know more about the Advance Data Residency Microsoft docs that support this discussion: `https://learn.microsoft.com/en-us/microsoft-365/enterprise/m365-dr-overview?view=o365-worldwide#table-2-durable-commitments-on-data-location-by-microsoft-365-service`



Before creating a new workspace, check whether an existing LAWS is already integrated with Defender for Cloud or other security services, as your workspace should be the same as the one already set up and being used for Defender for Cloud. You may also have a LAWS if you set up insights in the Entra ID Conditional Access portal. This is a common finding. You do not want to duplicate ingestion and cost. Either use existing or reroute logs to one location.

Also note that Microsoft Sentinel follows a **1:1 mapping**; only one Sentinel instance can be enabled per LAWS.

Enabling Microsoft Sentinel

Once your LAWS is provisioned as per the previous steps, proceed with the following:

1. Search for Sentinel in the Azure portal search bar. Navigate to **Microsoft Sentinel | + Create**.

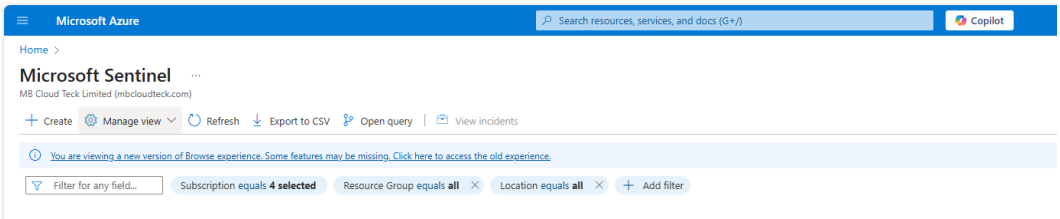


Figure 8.3: Searching for Sentinel in Azure

2. Select your LAWS and click **Add** to deploy Sentinel.

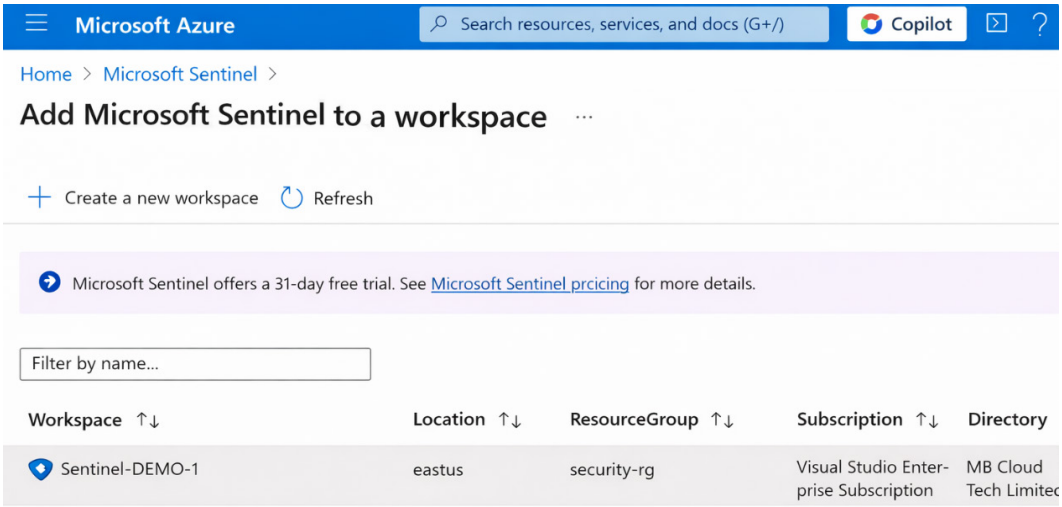


Figure 8.4: Adding Sentinel to a LAWS

Welcome to your SIEM! You've now deployed Microsoft Sentinel. You can now navigate to your new Sentinel instance to start connecting data sources, building analytics rules, and automating response workflows.

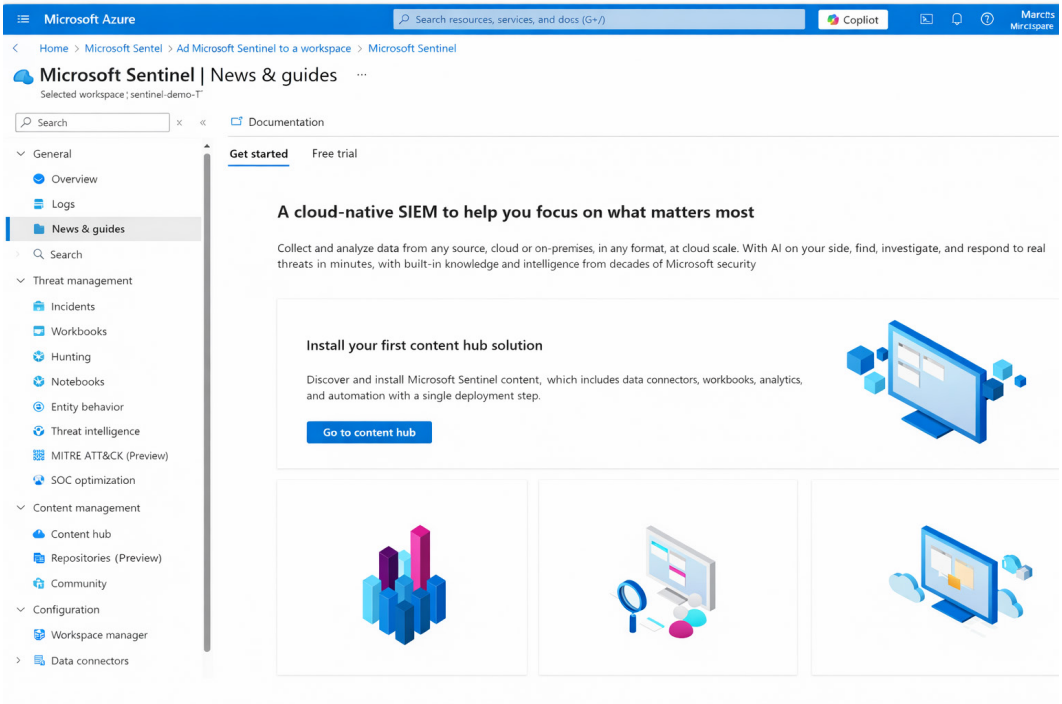


Figure 8.5: A fresh Microsoft Sentinel instance

Core Sentinel initial setup settings

Before configuring deeper features, start by enabling basic monitoring and cost-control measures. This helps you avoid unexpected charges and ensures visibility from day one.

For these steps, you will need the correct access rights. We recommend that during the build (non-prod), you have *Security Administrator* allow Sentinel and connector access, and *Owner* over the subscription. Hopefully, you set up the previous access packages and roles in *Chapter 6*. Go ahead and activate the required roles. Let's see how to set up a daily ingestion cap:

1. In the Azure portal, navigate to **Home | Monitor | Logs or Home | Log Analytics Workspaces.**
2. Select the LAWS that's connected to Microsoft Sentinel. Then, select **Workspace settings.**

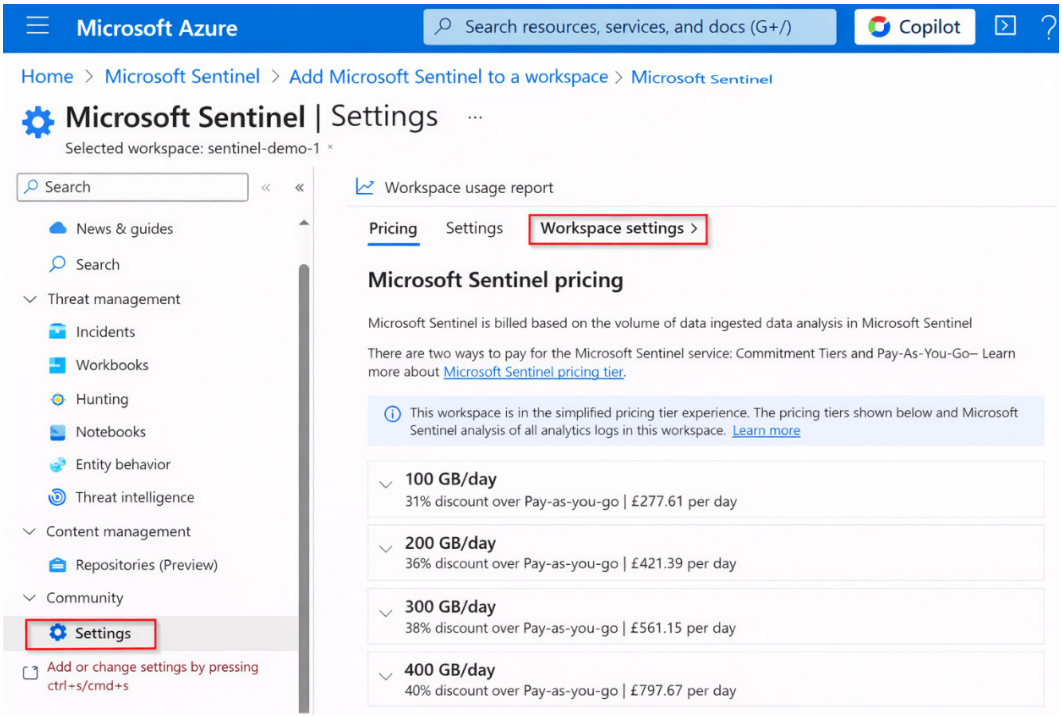


Figure 8.6: Microsoft Sentinel commitment tiers

3. In the workspace menu, click on **Usage and estimated costs**, as shown in *Figure 8.7*.

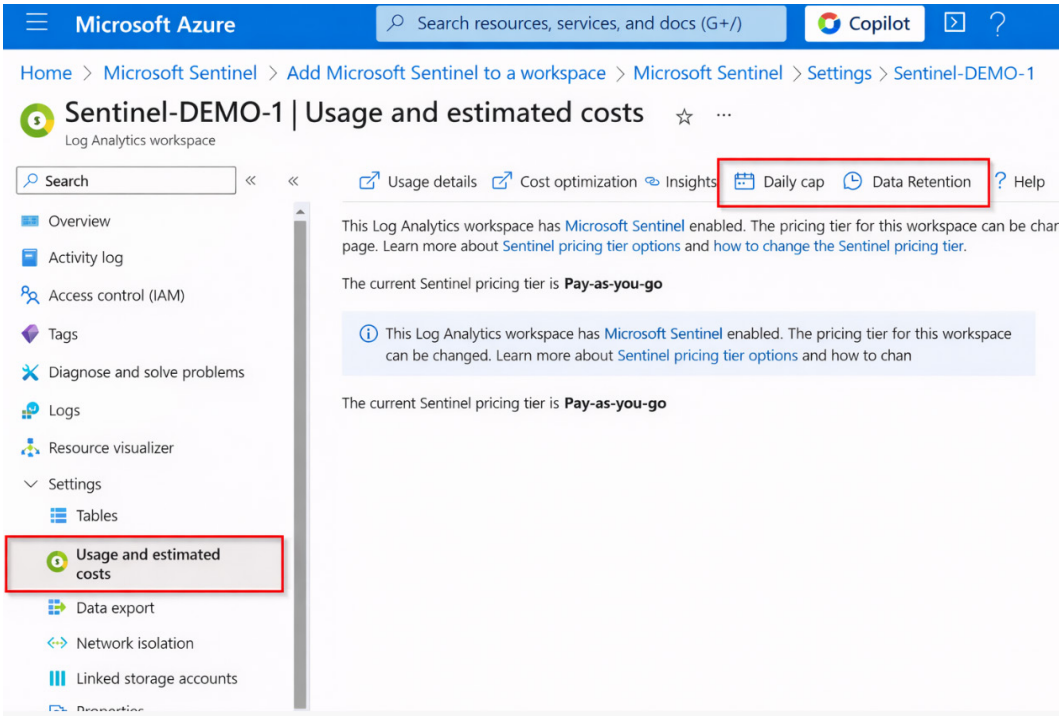


Figure 8.7: Setting daily ingestion caps for Sentinel

4. Toggle **Daily cap** to **ON**. Set your desired daily data ingestion limit in gigabytes (e.g., enter `10` for a 10 GB/day cap), as shown in *Figure 8.8*.

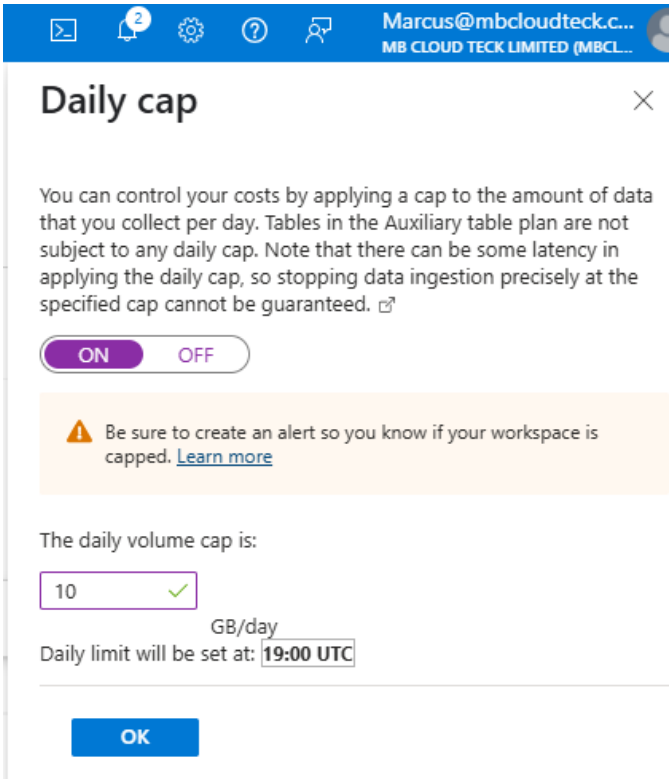


Figure 8.8: Cap configuration setting



Start with a conservative limit during early stages and adjust as your data volume and use cases grow.

As you can see below, the data retention slider can be moved from 30 days upwards to many years.

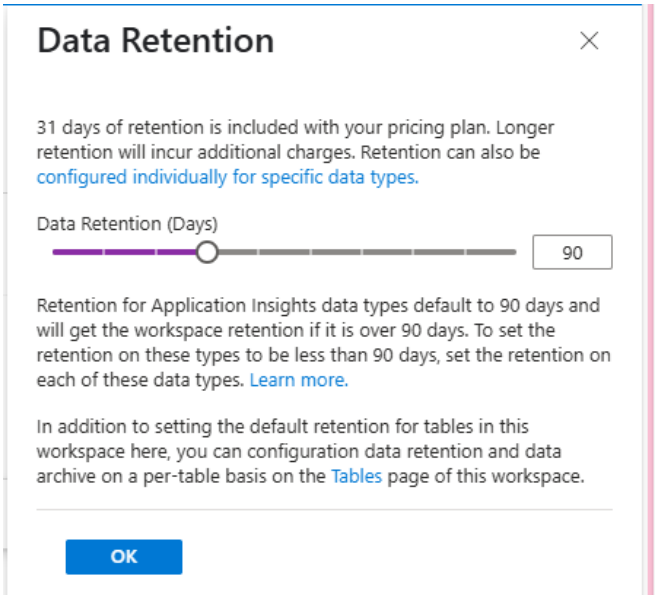


Figure 8.9: Data retention setting

With your daily ingestion cap in place, the next step is to enable diagnostic logging. This ensures that operational events related to your LAWS are captured. It helps with visibility, auditing, and troubleshooting.

Enabling diagnostic logs in your LAWS

Next, let's configure **diagnostic logs** for your LAWS by following these steps:

1. In the Azure portal, select your LAWS. In the left-hand menu, scroll down and click **Diagnostic settings**.

2. Click on **+ Add diagnostic setting** and enter a descriptive name for the setting.

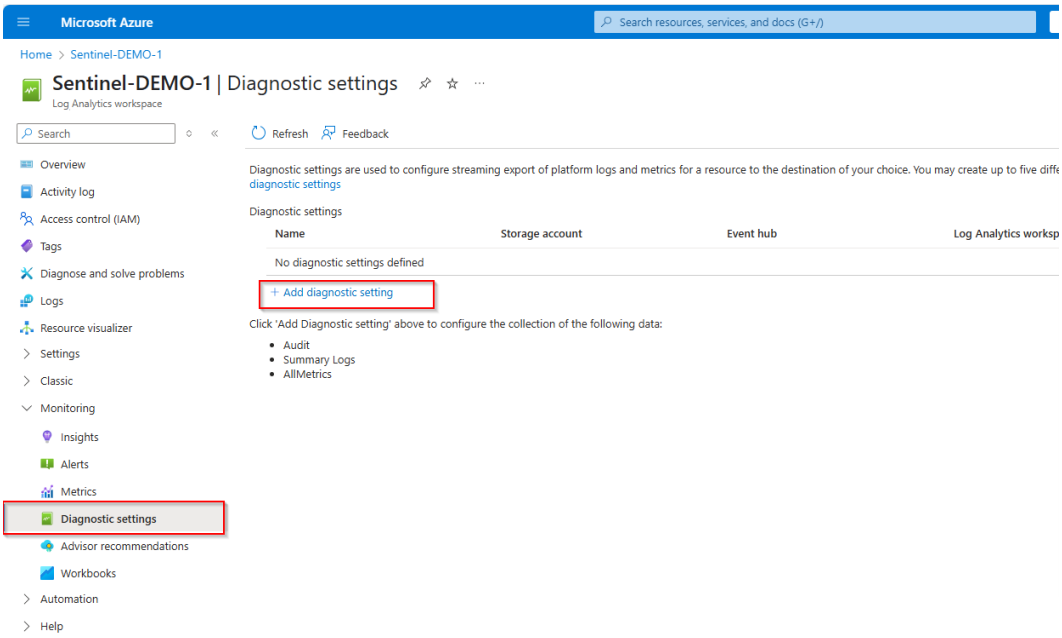


Figure 8.10: Adding diagnostic settings

3. Choose logs to collect from the following categories:

    - **AllMetrics**
    - **AuditLogs**
    - **WorkspaceHealth**
    - **DataIngestionStatus**

4. Check **Send to Log Analytics workspace** and choose the same or another workspace where the log diagnostics should be stored.

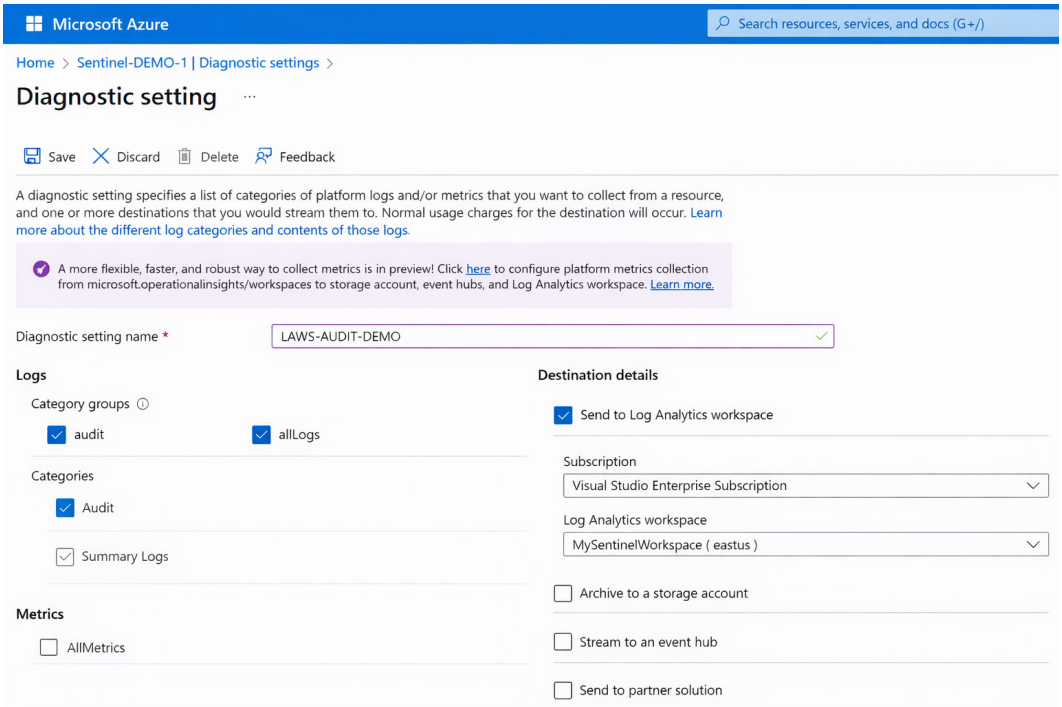


Figure 8.11: Diagnostic settings configuration

5. After saving the diagnostic setting, return to the **Diagnostic settings** page.
6. Make sure the setting appears in the list and that the relevant categories (e.g., **AllMetrics**, **WorkspaceInsights**, etc.) are enabled.

With diagnostic settings configured and essential logs flowing into your workspace, it's time to expand your visibility, specifically around user behavior and identity-based threats. This is where **UEBA** comes in. UEBA is a built-in capability in Microsoft Sentinel that adds behavioral context to your data. It helps you detect anomalies such as unusual logins, privilege misuse, or insider threats by learning what "normal" looks like and flagging deviations.

Let's discuss how to enable and start using UEBA in your environment.

UEBA

UEBA in Microsoft Sentinel helps detect insider threats, compromised accounts, and unusual activity by analyzing identity-related telemetry and building behavioral baselines. You can find it in the Azure portal. Go to **Microsoft Sentinel | Settings | User and Entity Behavior Analytics (UEBA)**.

UEBA ingests signals from Entra ID, Microsoft Defender, and other identity-related sources. It establishes a baseline for normal user and entity behavior. It also identifies anomalies such as impossible travel, including multiple logins from geographically distant locations within a short time, unusual sign-in locations from unfamiliar countries or IP ranges, and rare role assignments with privilege escalations that deviate from typical patterns. These detections enrich incidents, trigger analytics rules, and support deep-dive investigations with behavioral context.

Follow these steps to turn on UEBA:

1. Go to `https://portal.azure.com` and log in with an account that has access to Microsoft Sentinel.
2. In the left-hand menu, search for and select **Microsoft Sentinel**. Select the workspace where Sentinel is enabled.
3. In the left-hand menu under **Configuration**, click **Settings**. Then, click on **User and Entity Behavior Analytics**.

4. On the UEBA page, toggle the switch to **On** to enable UEBA, as shown in *Figure 8.12*. Click **Apply** to confirm the change.

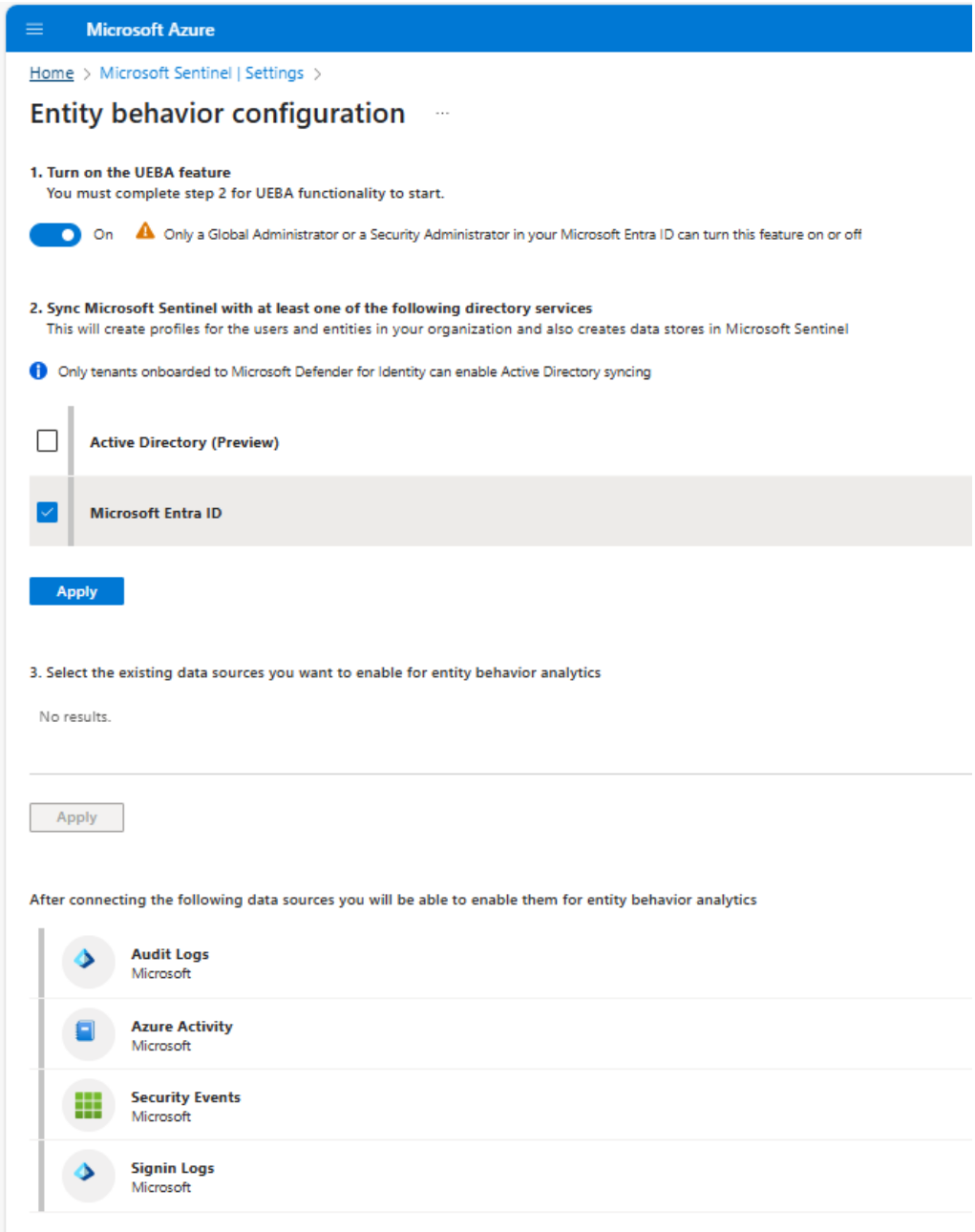


Figure 8.12: UEBA configuration



UEBA works best when Microsoft Sentinel has access to identity-related logs such as Entra ID sign-ins, audit logs, and Defender telemetry.

UEBA will begin ingesting signals from connected identity and telemetry sources such as Entra ID and Microsoft Defender. No additional configuration is required unless specific sources need to be onboarded.

Once UEBA is enabled, Sentinel doesn't just correlate raw events into incidents—it begins learning "normal" behavior for users and entities over time. With that baseline in place, UEBA then surfaces anomalies: focused signals where current activity deviates from expected patterns, such as a user who typically logs in from the UK during business hours suddenly authenticating from a new location at 02:00 and accessing unfamiliar resources.

Now that UEBA data is connected, we need to monitor for anomalies.

Anomalies in Sentinel

The **Anomalies** feature works alongside UEBA to surface real-time deviations from expected behavior, helping you catch threats that don't follow known attack patterns. You can find it by following this pathway in the Azure portal: **Microsoft Sentinel | Settings | Anomalies**.

This lists anomalies with details such as **user**, **timestamp**, **anomaly type**, **risk score**, and **severity**. **Anomalies** integrate with alert rules and incident timelines to support active investigations. Enabling the feature also helps you proactively detect subtle or emerging threats that traditional rule-based detection may miss.

Anomalies are ML-driven alerts that flag activity deviating from a normal baseline—for example, `alex.w@contoso.com` suddenly logging in 20× more than usual at 02:14 UTC, producing a **High** anomaly score for *unusual login volume*.

As you enable advanced capabilities such as UEBA and anomaly detection, it's important to understand what data is being used and how it's being handled.

This is important because UEBA and anomaly detection are only as trustworthy as the data they learn from, and that data can carry security and privacy implications.

If the underlying signals are incomplete, noisy, or biased (e.g., missing identity logs, inconsistent device telemetry, or shared service accounts), the "baseline" becomes inaccurate, and anomalies become either meaningless noise or missed risk.

Analysts will act on anomaly output. Knowing which sources feed the models and how they're processed supports defensible triage decisions and better tuning, especially when you need to explain "why this was flagged" to stakeholders. Understanding the data pipeline is what turns UEBA/anomalies from a "black box feature" into a governed, auditable, and effective capability.

Microsoft Sentinel provides built-in transparency around data usage, processing, and compliance.



How Microsoft uses your data

Sentinel includes a dedicated section that outlines how your data is used. You can find this by following this pathway in the Azure portal: **Microsoft Sentinel | Settings | How we use your data**. The page will give you details on where and how data is processed and stored, how Microsoft complies with standards such as GDPR and ISO, what data is used for capabilities such as UEBA, analytics rules, and threat detection, and the privacy and security safeguards in place throughout the process.

Now that UEBA is enabled and enriching your investigations with behavioral context, it's time to ensure that Microsoft Sentinel is connected to the new SecOps platform, Defender XDR.

Connecting the unified platform

As part of the baseline setup, integrate Microsoft Sentinel with Microsoft Defender XDR and treat your primary Sentinel workspace as the central point for security telemetry within the tenant. This includes Microsoft-native, service-to-service connectors such as Microsoft Entra ID and Defender XDR which are tenant-bound by design and can be enabled without additional log forwarding agents or bespoke onboarding effort.

Once the connection is in place, Sentinel can correlate Defender incidents with other log sources, giving your SOC a single, consistent view for detection, investigation, and response. You can start configuring these integrations and managing the unified environment from the Microsoft Defender XDR portal, where many of the core security controls and cross-product settings are now centralized.

From the Microsoft Defender XDR portal, follow these steps:

1. Go to the unified SecOps portal at `https://security.microsoft.com` and, from the left-hand navigation pane, select **Settings**. Choose **Microsoft Sentinel** to view connected workspaces and available configurations.

   You can also use the top banner in the XDR portal if shown. It will take you to the same place.

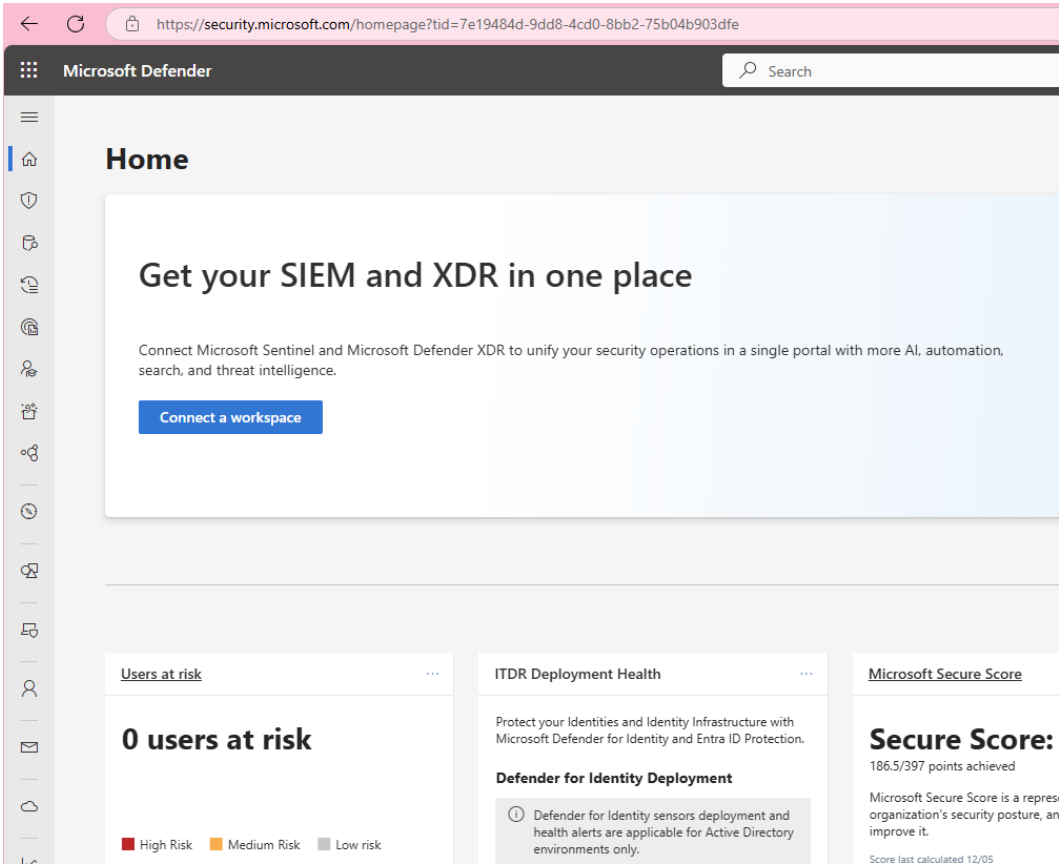


Figure 8.13: Unified SecOps portal

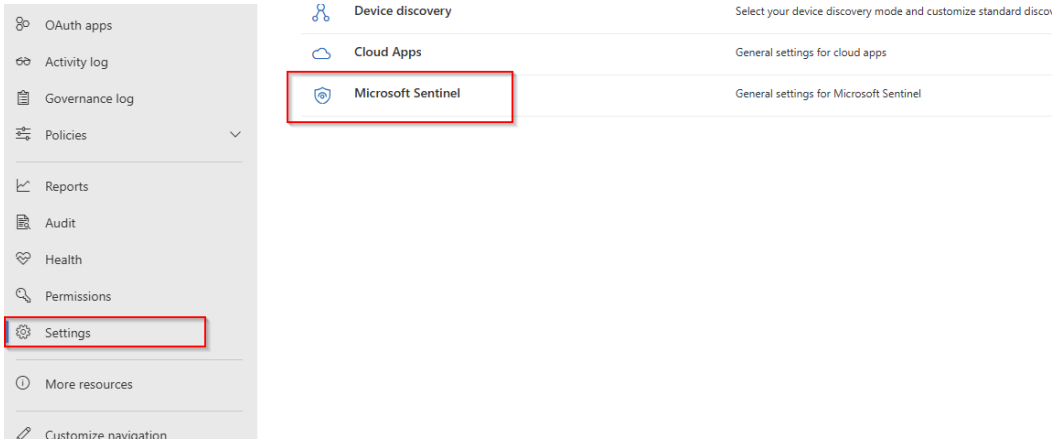


Figure 8.14: Enabling Microsoft Sentinel in the unified SecOps portal

2. Select the workspace you want to connect by checking the box next to its name (e.g., **MySentinelWorkspace**), then click **Next**.

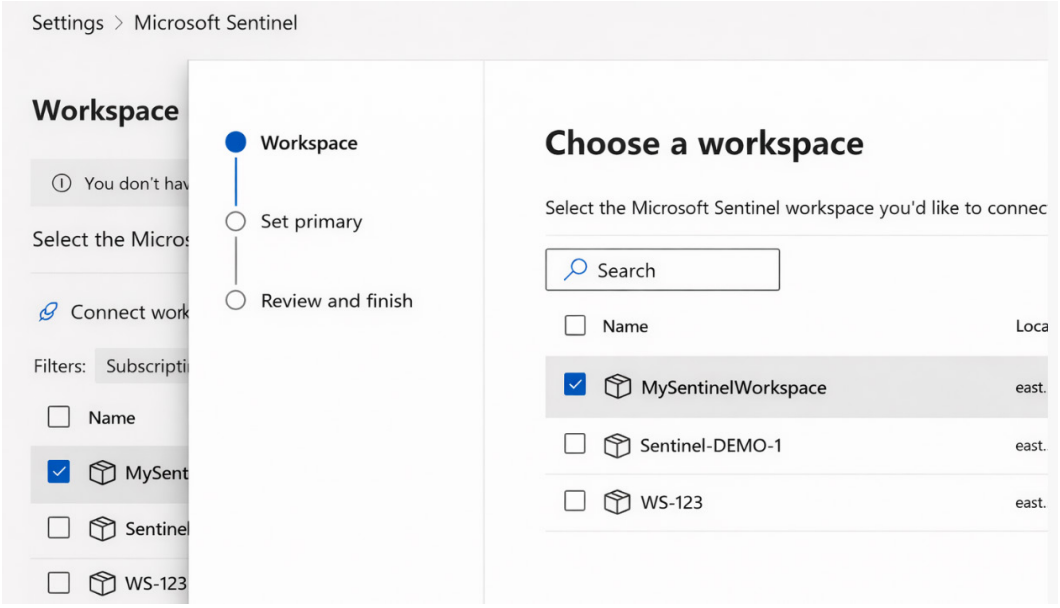


Figure 8.15: Connecting the workspace to the unified SecOps portal

3. Use the dropdown to choose the workspace (e.g., **Sentinel-DEMO-1**) that will serve as your primary workspace. This is the one that will receive alerts and incidents from Defender XDR. Then, click **Next**.

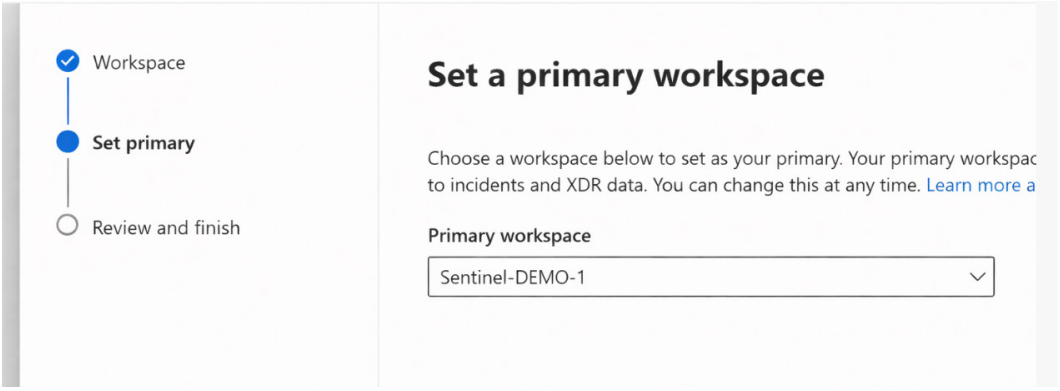


Figure 8.16: Setting up the primary workspace

4. Review your selection and the impact of enabling the connection:

    - Only alerts tied to incidents will be sent to Sentinel
    - Sentinel will assign roles as needed (e.g., Microsoft Sentinel Contributor)
    - Microsoft Security incident creation rules for alerts already handled by Sentinel will be disabled to avoid duplication

    Then, click **Next**.

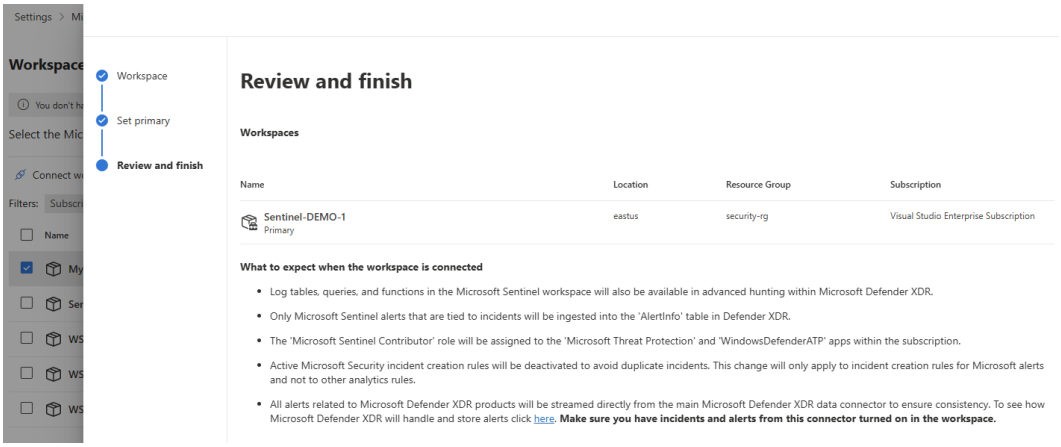


Figure 8.17: Reviewing and finishing the workspace connection

5. Click **Connect** on the popup to confirm the workspace connection to proceed.

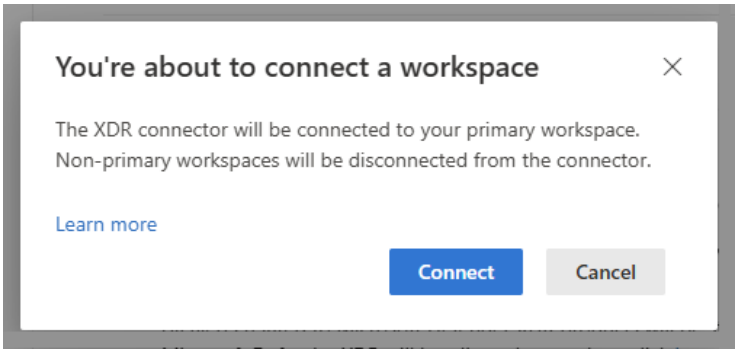


Figure 8.18: Final step before connecting the workspace to the unified SecOps portal

You'll see confirmation that your Sentinel workspace is successfully connected. From here, you can view incidents across Sentinel and Defender, hunt across synced data, and manage related settings.

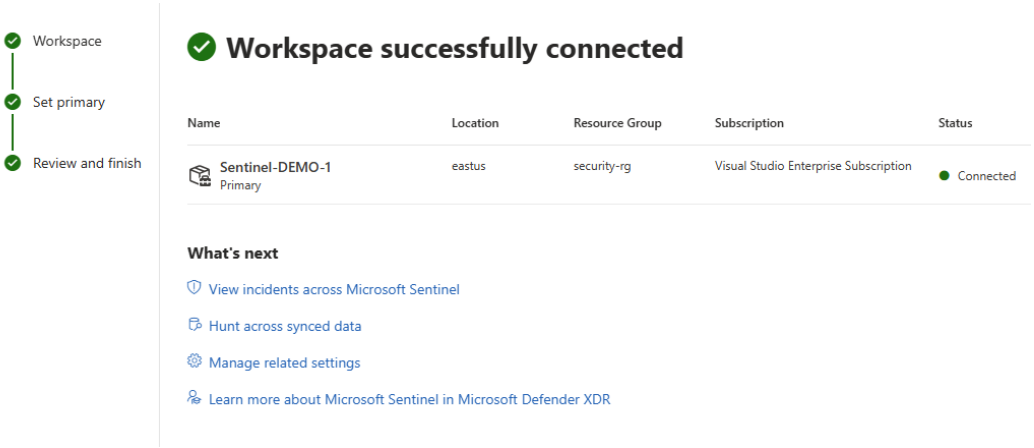


Figure 8.19: Workspace connected to the unified SecOps portal

If the connection is successful, your portal will look like *Figure 8.25*:

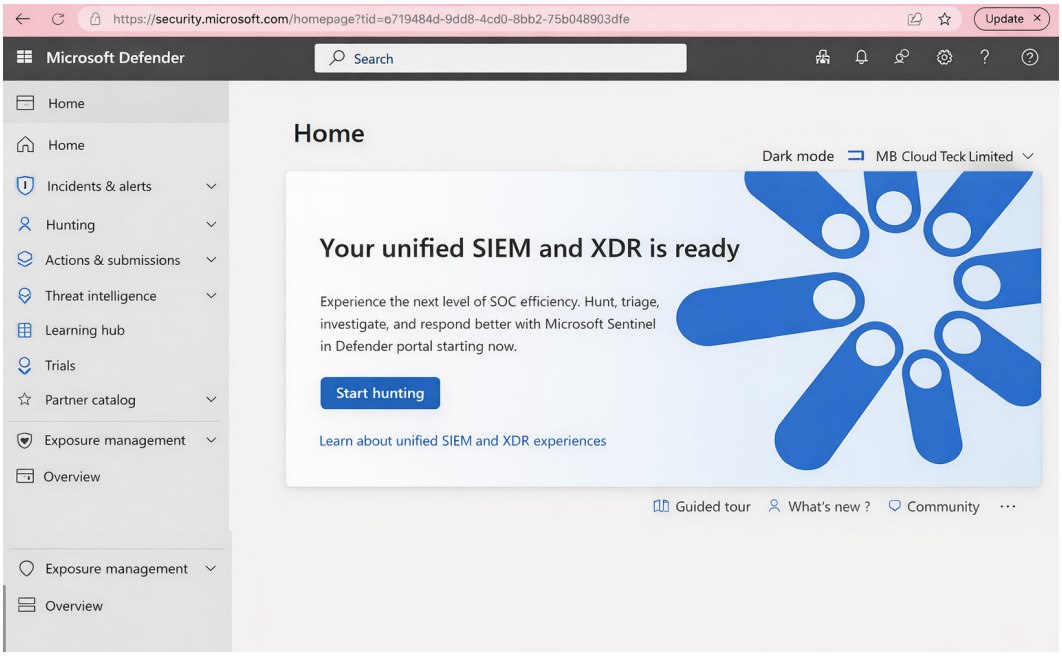


Figure 8.20: Unified SecOps portal view once the Sentinel workspace has been connected

With your workspace connected and the platform integrations in place, you now have a secure and unified foundation to build on.

Optional: Multi-tenant management

If you're working in a multi-tenant environment, use the tenant switcher at the top right of the Microsoft Defender portal to manage integrations across tenants. Switching tenants and carrying out tasks such as threat hunting against multiple tenants are different. Currently, MTO has limited capability, and you will have to switch to use a lot of its functions. To switch tenants, click on your tenant's name and select **Switch tenant** or **Multi-tenant management**, as shown in *Figure 8.21*.

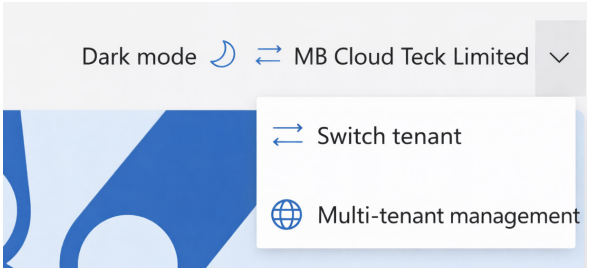


Figure 8.21: MTO view in the unified SecOps portal

Microsoft now fully acknowledges the operational need for organizations managing multiple tenants to implement and oversee security across all of them simultaneously. Rather than treating each tenant as an isolated security island, the direction is shifting toward unified, scalable security management, which enables security teams to apply policies, monitor threats, and respond to incidents across their entire tenant landscape in a cohesive way. *Table 8.8* shows the list of some of the currently supported capabilities to perform against multiple tenants via the MTO portal.

Capability	Description
Incidents and alerts | Incidents	Manage incidents originating from multiple tenants.
Incidents and alerts | Alerts	Manage alerts originating from multiple tenants.
Cases	Manage cases originating from multiple tenants.
Hunting | Advanced hunting	Proactively hunt for intrusion attempts and breach activity across multiple tenants at the same time.
Hunting | Custom detection rules	View and manage custom detection rules across multiple tenants.
Assets | Devices | Tenants	For all tenants and at a tenant-specific level, explore the device counts across different values such as device type, device value, onboarding status, and risk status.
Endpoints |Vulnerability Management | Dashboard	The Microsoft Defender Vulnerability Management dashboard provides both security administrators and security operations teams with aggregated vulnerability management information across multiple tenants.
Endpoints | Vulnerability management | Tenants	For all tenants and at a tenant-specific level, explore vulnerability management information across different values such as exposed devices, security recommendations, weaknesses, and critical CVEs.
Configuration | Settings	This lists the tenants you have access to. Use this page to view and manage your tenants.

Table 8.8: Defender XDR Portal Options

Collecting data and deploying solutions

With the planning and setup phases behind you, it is time to shift into execution. The next critical step is to start collecting security data by deploying Microsoft Sentinel solutions and configuring service-to-service integrations. This forms the operational backbone for advanced analytics, automation, and real-time threat detection.

Deploying content hub solutions

After setting up your data connectors, head to the content hub in the Azure portal to explore and deploy solution packs. These curated packs contain prebuilt workbooks, analytics rules, playbooks, and other content designed to accelerate time-to-value and provide actionable templates tailored to your connected data sources. These are designed to accelerate time-to-value by aligning with best practices and mapping directly to the data sources you've connected to, so you're not starting from scratch.

We recommend prioritizing these core service-to-service connectors presented in *Table 8.9* during your initial setup:

Solution area	**Description**
Microsoft 365	This core collaboration and productivity platform enables visibility into email, SharePoint, Teams, and so on
Microsoft Entra ID	Identity and access management is crucial for detecting suspicious sign-ins and access anomalies
Defender for Endpoint	Endpoint protection signals detect malware, lateral movement, and device-based risk
Defender for Identity	Identity-based threat detection flags domain controller activity, pass-the-hash attacks, and so on
Defender for Office 365	Email threat intelligence detects phishing, spoofing, and malicious links or attachments
Defender for Cloud Apps	Visibility into SaaS and cloud usage helps detect shadow IT, risky app behavior, and data exfiltration

Table 8.9: Service To Service Sentinel Connectors



Look for solution packs marked **Featured** or **Essentials**, as these provide high-impact, streamlined content ideal for initial deployments.

So back to connecting these Service to Service connectors.

Select **Defender XDR** via the checkbox and click **Install** on the top bar. Here, we installed Microsoft Defender XDR. You can see this via the green check mark:

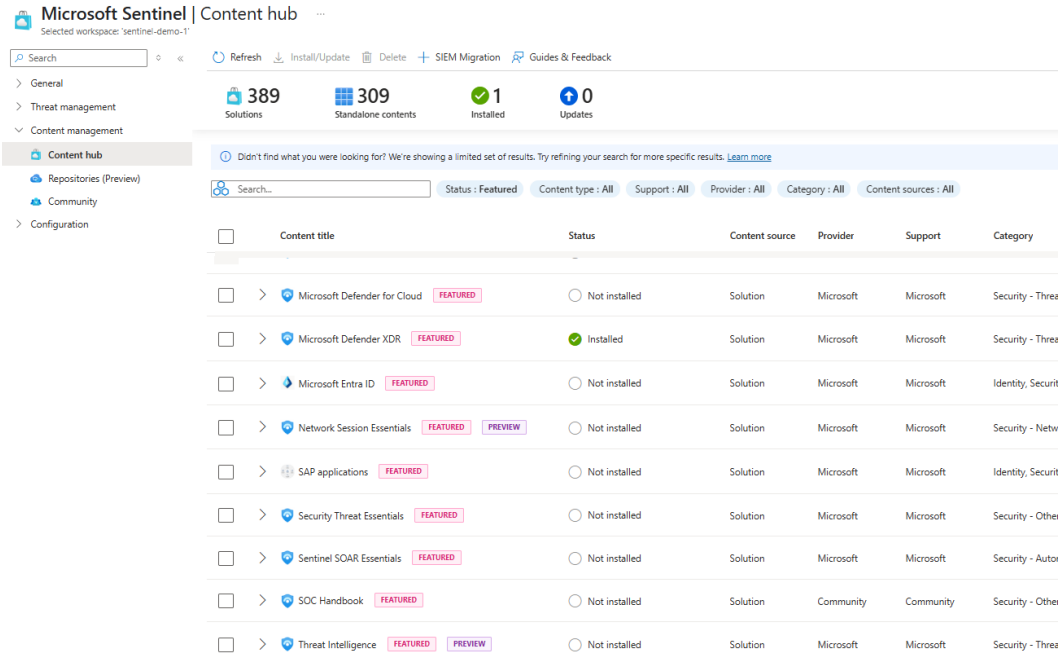


Figure 8.22: Content hub solutions for Microsoft Sentinel



If the solution pack includes a data connector, you'll see a **Manage** option once installed. Use this to complete the connector setup and ensure telemetry is flowing into your workspace.

Where the content pack includes a data connector, once installed, you can manage the connector as shown in the following steps.

Step 1: Access the Defender XDR solution

To access the Defender XDR solution, follow these steps:

1. After installing the solution from **Content hub**, click **Manage** on the Microsoft Defender XDR solution page.

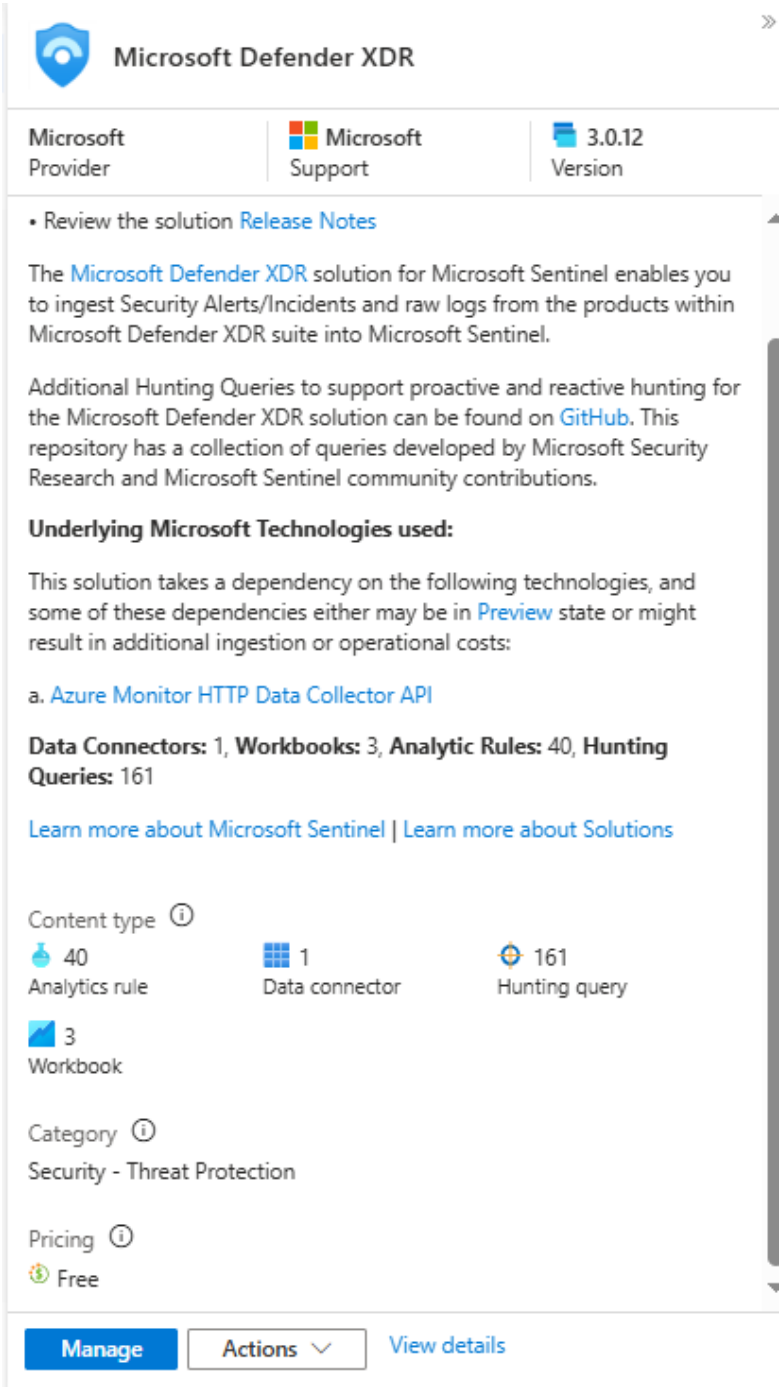


Figure 8.23: Microsoft Defender XDR data connector

This opens the connector blade where you can configure, monitor, and fine-tune the integration.

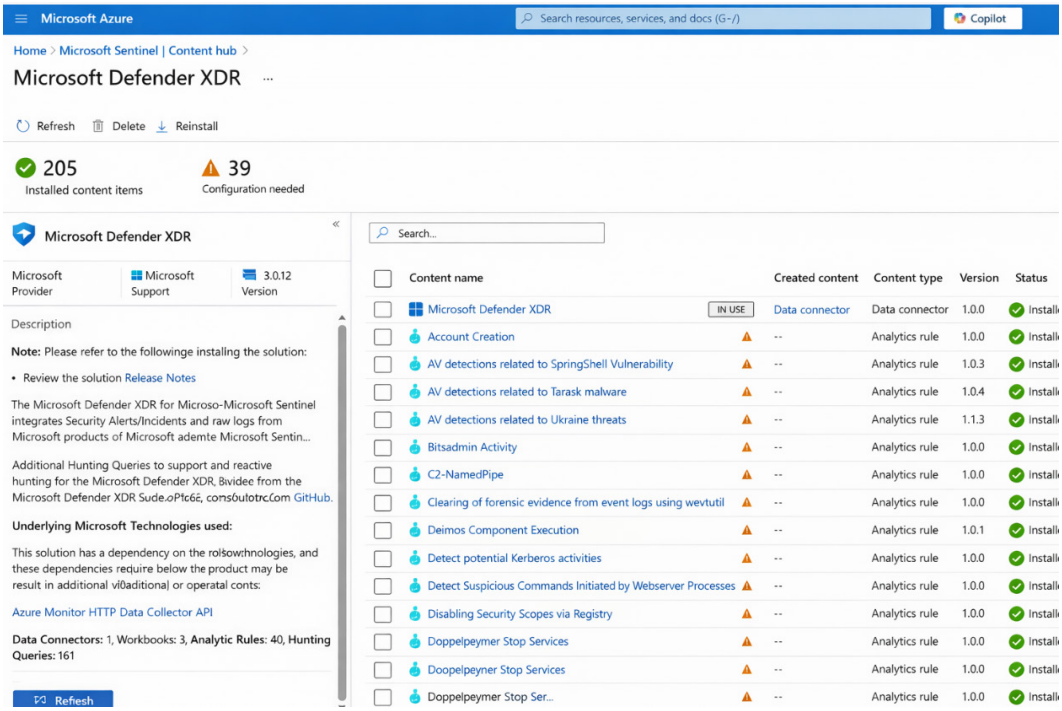


Figure 8.24: Microsoft Defender XDR data connector artifact components

Take a look at the Defender XDR Content Hub Overview and the relevant products included that can be connected. You will need to click into it to configure.



Figure 8.25: Microsoft Defender XDR sub-product components

2. Click on **Open connector page**, which will take you to the main connector dashboard.

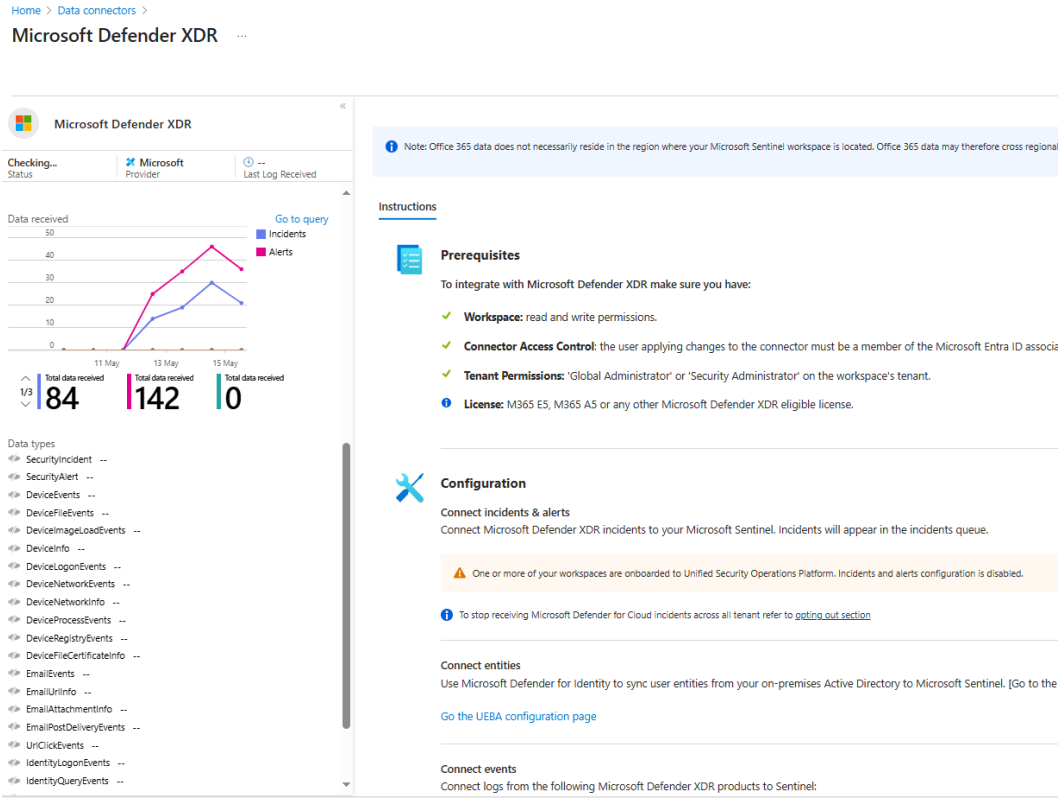


Figure 8.26: Microsoft Defender XDR data connector showing data being ingested

Table 8.10 explains what the data on the dashboard represents:

Section	Description
Connector overview panel (top left)	Shows connection status (e.g., **Connected**, **Validating**), **Provider** (Microsoft), and **Last Log Received** (timestamp of the latest log). Click **Open connector page** to access the main dashboard.
Data received graph (pink/blue lines - middle)	Displays log ingestion trends for the past 7 days. The blue line shows the incident volume, and the pink line shows the alert volume. The log confirms whether data is flowing and patterns are emerging.

Section	Description
Data types and table selection (bottom left)	Lists log types (**SecurityIncident**, **DeviceEvents**, **EmailEvents**, etc.). You can enable or disable specific tables to manage scope and cost. Each maps to Defender components: *Defender for Endpoint*: **DeviceInfo**, **DeviceProcessEvents**, and **DeviceFileEvents** *Defender for Identity*: **IdentityLogonEvents** and **IdentityQueryEvents** *Defender for Office 365*: **EmailEvents** and **EmailAttachmentInfo** *Defender for Cloud Apps*: **CloudAppEvents** and **ShadowITDetection**
Configuration panel (bottom right)	Connect incidents and alerts to the Sentinel incident queue: connect entities for UEBA and event tables selectively. Includes links to official documentation, UEBA configuration page, and licensing requirements.
Prerequisites for integration (top right)	This section includes **Workspace** (read/write permissions), **Connector Access Control** (member of Entra ID admin group), **Tenant Permissions** (**Global** or **Security Administrator**), and **License** (M365 E5, A5, or Defender XDR).

Table 8.10: Details of data on the dashboard



Start with core service-to-service connectors such as Azure activity logs, Microsoft 365 audit logs, and Defender products to gain immediate visibility. Most offer one-click setup and provide a strong foundation for detections and automated responses in Microsoft Sentinel.

With your core solutions deployed and data flowing into Microsoft Sentinel, the next step is to tailor what's been installed to fit your environment.

Step 2: Customizing and optimizing

After installing content from various content hub solutions, you need to tailor the default configurations to match your organization's security and compliance requirements. This could include modifying analytics rules to fine-tune alert thresholds and reduce noise, adjusting workbooks and dashboards for relevant visibility across teams, and aligning playbooks and automation rules with your incident response workflows.

Let's walk through a practical example using one of the rules from the **Microsoft Defender XDR** solution pack:

1. From **Content hub**, install the solution and review the included components. Here is an example:

    - xx Analytics Rules
    - x Workbooks
    - xx Hunting Queries
    - x Data Connector

    Before installing, explore the content of each solution, including rule templates, workbooks, and connectors via the side panel in **Content hub** before installing.

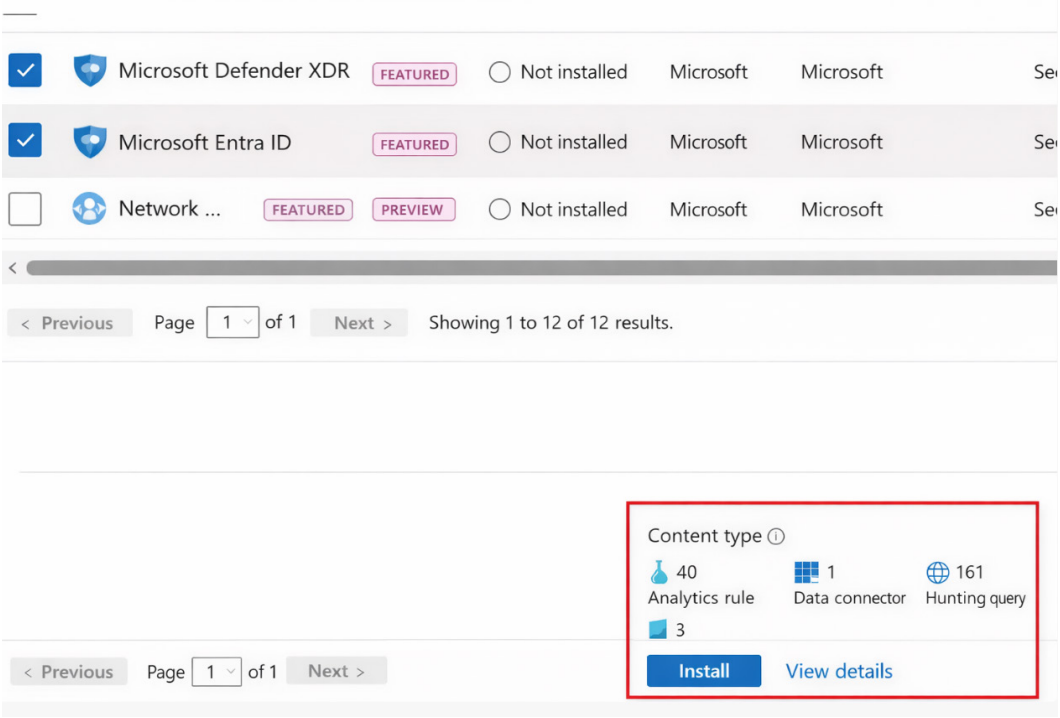


Figure 8.27: Microsoft Sentinel Content hub solution components – multiple

You can explore the content of each solution, including rule templates, workbooks, and connectors, via the side panel in **Content hub** before installing.

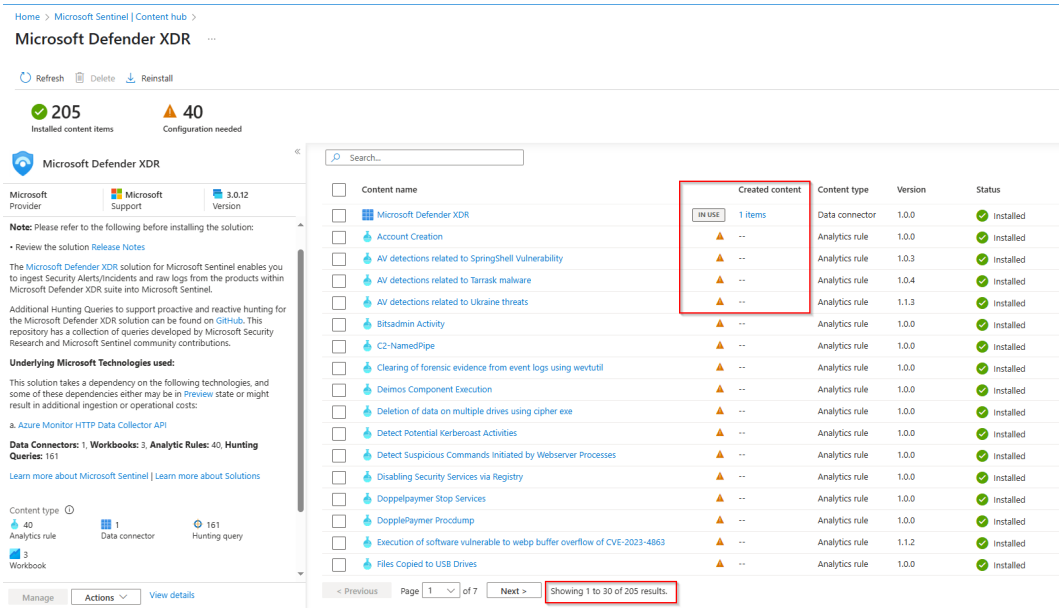


Figure 8.28: Microsoft Sentinel Defender XDR components

To explore some content, let's focus on a high-impact rule designed to detect potential **data exfiltration** via USB devices.

2. Navigate to the Defender XDR solution's analytics rules and choose the rule titled **Files Copied to USB Drives**. Then, click **Create rule** to begin configuration and review the logic.

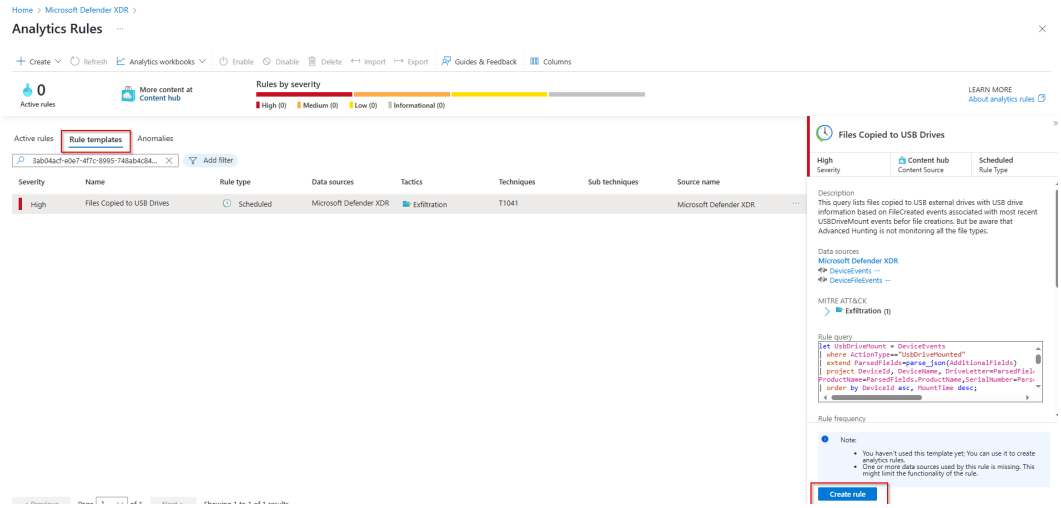


Figure 8.29: Microsoft Defender XDR built-in analytic rules

3. Begin reviewing the configuration and make any edits to the rule logic.

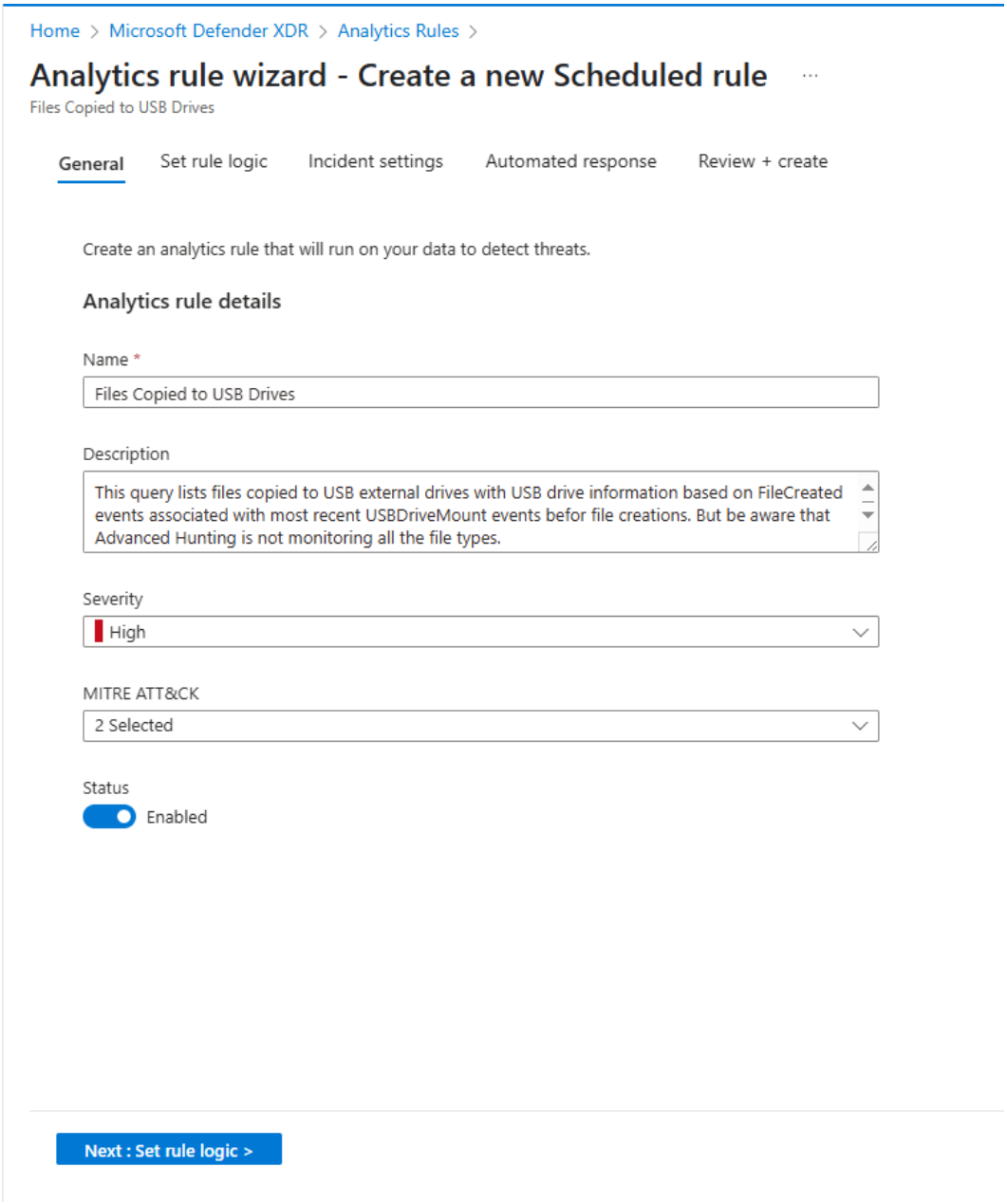


Figure 8.30: Microsoft Defender XDR rule creation

Table 8.11 explains each component within the **General** tab under **Analytics rule wizard**:

Field	Value/Description
Name	`Files Copied to USB Drives`
Description	Rule Description
Severity	**High** – Indicates potential insider threat, data theft, or policy violation.
MITRE ATT&CK	Two techniques selected: • TA0009 (Collection) • TA0010 (Exfiltration)
Status	**Enabled** – The rule will be active immediately and execute on the defined schedule.

Table 8.11: Components of General Tab

4. Now, move to the **Set rule logic** tab and analyze the detection logic below it, as shown in *Figure 8.31*.

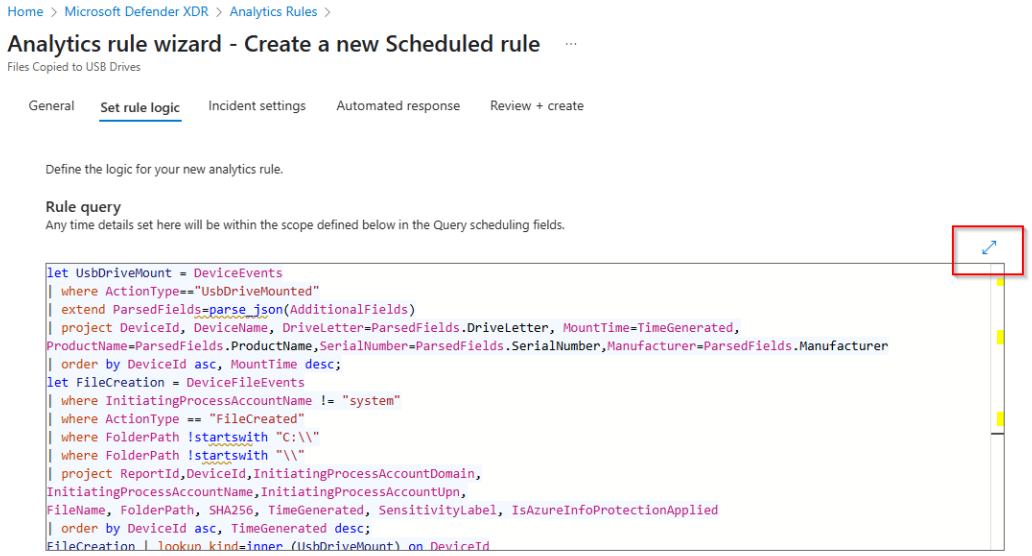


Figure 8.31: Microsoft Defender XDR KQL analysis

Here, the detection is authored using **Kusto Query Language** (**KQL**). This is where detection engineering comes into play. The rule does the following:

- Pulls from `DeviceEvents` and `DeviceFileEvents`
- Correlates **USB mount activity** with **file creation** to identify potential data exfiltration
- Filters out system-initiated actions
- Highlights sensitive paths such as `C:\` and `Users\`, and flags unusual write activity

It also enriches the detection with device-level metadata such as `DeviceName`, `SerialNumber`, and `Manufacturer`, providing the analyst with **full context** in a single alert. The precision of this logic, which includes joins, filtering, and project-renaming, is what separates noisy rules from operationally valuable ones. It's designed not just to detect, but to explain what happened, where, and why it matters.

So far, we have explored some content hub content and broken down an example. Now, let's take a look at some other features to complement all of the available content.

The backbone of Microsoft Sentinel

Let's take a deeper look at the core components that form the foundation of Microsoft Sentinel. Approaching them will help you gain the clarity needed to explain these features confidently to your customers.

Introduction to KQL and detection rules

The easiest way to think about KQL is to compare it to working with multiple **Excel spreadsheets**. Imagine you have thousands of rows of security data across different tables. KQL is like a supercharged filter and analysis engine. It helps you process that data step by step, table by table, column by column.

Each pipe symbol (|) in KQL acts like a stage in a data transformation process. It's similar to saying, *Start with this dataset → filter out what I don't need → select the columns I care about → sort or summarize the results.*

If you've worked with **PowerShell, Excel formulas**, or **SQL**, you'll feel right at home.

Here's a simple example to show how it works:

Step	Action	KQL syntax	Explanation
1	Start with a table	`SecurityEvent`	Think of this like opening a full Excel sheet with raw event data—every row is a log entry.
2	Filter for failed logons	`SecurityEvent | where EventID == 4625`	This narrows down the dataset to only failed login attempts, just like applying an Excel filter.
3	Select the relevant columns	`| project TimeGenerated, AccountName, Computer`	This is like hiding all unnecessary columns to focus only on the data you care about.
4	Sort the results	`| order by TimeGenerated desc`	You sort the failed logons in descending order by time—newest first—just like sorting in Excel.
5	Final query	`SecurityEvent | where EventID == 4625 | project TimeGenerated, AccountName, Computer | order by TimeGenerated desc`	Clean, focused dataset ready for analysis. No noise, just insight.

Table 8.12: Examples for KQL formulas



Pro tip: For the ultimate guide to KQL, review Rod Trent's KQL book here: `https://aka.ms/MustLearnKQL`.

Understanding KQL syntax basics

The query structure of KQL queries is a series of statements connected by pipes (|). Each statement processes data and passes it to the next step. Keywords and operators are case-insensitive (e.g., `where`, `WHERE`), and field names and string values are case-sensitive. Always write clean, readable queries by formatting them with line breaks after each pipe (|).

For example, the following is a basic KQL query:

```
SecurityEvent
| where EventID == 4625
| project TimeGenerated, AccountName, Computer
| order by TimeGenerated desc
```

This query retrieves failed logon events, selects key columns, and orders the results by time.

Essential KQL functions you need to know

Let's shift focus to a practical guide that walks through other essential functions you need to know, starting with the basics of the KQL syntax.

Core operators and their use cases

In KQL, core operators are used to filter (where), shape (project, extend), aggregate (summarize), sort (order by), and correlate (join) data, enabling analysts to efficiently query, analyze, and derive security insights from large volumes of log data

- Filtering data with `where`: This function helps extract rows that meet specific conditions

  Syntax:

  ```
  TableName
  | where Condition
  ```

  Examples:

  ```
  | where EventID == 4625: Filter for failed logon events.
  | where AccountName startswith "admin": Retrieve accounts starting
  with "admin".
  ```

- Selecting columns with `project`: This function limits the output to specific columns for clarity and performance

  Example:

  ```
  SecurityEvent
  | project TimeGenerated, AccountName
  ```

- Adding new columns with `extend`: This creates calculated columns for better analysis

  Example:

  ```
  | extend AccountNameLength = strlen(AccountName)
  ```

- Aggregating Data with summarize: This groups and calculates summary statistics

  Example:

  ```
  | summarize TotalEvents = count() by EventID
  ```

- Sorting data with `order by`: This arranges data in ascending or descending order

  Example:

  ```
  | order by TimeGenerated desc
  ```

Advanced filters and expressions

Advanced filters and expressions use functions, conditional logic, pattern matching, and time-based operations (such as parse, extract, case, has_any, and ago) to apply precise, context-aware filtering and enrich results beyond simple value matching.

- **Logical operators**: The function combines multiple conditions using `and`, `or`, and `not`

  Example:

  ```
  | where EventID == 4625 and AccountName contains "admin"
  ```

- **String operations**: This uses functions such as `startswith`, `contains`, `has`, and `endswith` to filter textual data

  Examples:

  ```
  | where Computer has "server"
  | where AccountName endswith "123"
  ```

Transforming data

Transforming data means reshaping and enriching records using operators like extend, project, parse, split, and mv-expand, so raw log fields become clean, consistent, analysis-ready data for detection, hunting, and reporting.

- **Parsing fields**: This extracts or manipulates structured fields within logs

  Examples:

  ```
  Using parse: | parse Message with "User=" Username ";"
  Using extract: | extend ExtractedField = extract("User=(\w+)", 1,
  RawData)
  ```

- **Time functions**: These group events into time intervals with `bin`

  Example:

```
| summarize count() by bin(TimeGenerated, 1h)
```

Aggregations

Aggregations use summarize (with functions like `count()`, `dcount()`, `avg()`, `min()`, `max()`, and percentiles) to group events and compute statistics over time, entities, or dimensions for trend analysis, anomaly spotting, and reporting.

Common functions include the following:

- `count()`: Total number of rows
- `sum()`: Total of a numerical column
- `avg()`: Average of a numerical column
- `min()` / `max()`: Minimum or maximum value

Grouping data

Grouping data is typically done with summarize … by (often paired with bin() for time) to cluster events by shared fields such as user, host, IP, or timeframe so you can analyse patterns and distributions at the right level of detail.

- Use `summarize` with by to group and calculate statistics

  Example:

```
| summarize TotalLogons = count() by AccountName
```

Joining and enriching data

Joining and enriching data uses operators like join, lookup, and union to correlate events across tables and add context (for example, identity, asset, or threat intel attributes), producing a single, investigation-ready view.

- **Joins**: This function combines datasets using `join`

  Syntax:

```
Table1 | join kind=inner (Table2) on KeyField
```

Example: Enrich failed logons with account details:

```
SecurityEvent
| where EventID == 4625
| join kind=inner (UserDetails) on AccountName
```

- **Lookup**: Simplifies enrichment for small datasets

  Example:

  ```
  | lookup kind=leftouter UserDetails on AccountName
  ```

Visualization

Visualization uses rendering commands such as render timechart, barchart, or piechart to turn query results into visual outputs that highlight trends, distributions, and anomalies for rapid analysis and reporting.

- **Adding visualizations**: Use render to display data as charts.

  Let's look at some examples.

  For a time chart, you can use a similar syntax to the following:

  ```
  | summarize count() by bin(TimeGenerated, 1h)
  | render timechart
  ```

  For a bar chart, use a similar syntax to the following:

  ```
  | summarize count() by AccountName
  | render barchart
  ```

Now that we understand some of the raw KQL functions, you may be able to decipher the intention of a KQL rule. Let's look at some rule logic.

Rule logic and entity mapping

Let's return to the analytics rule and break down what the **Rule Logic** section does. This part of the rule is essential for defining how detections are evaluated and how their results are interpreted. A key component here is entity mapping, which plays a crucial role in improving the quality and usability of alerts. By mapping entities such as account name, devices, users, files, and file hashes, you help Microsoft Sentinel understand what each piece of data represents. This context enables better correlation, investigation, and enrichment of alerts in both the investigation graph and other connected experiences.

For example, consider the following query:

```
| summarize count() by AccountName
| render barchart
```

This simple query aggregates results by account and visually represents them. But without proper entity mapping (within the rule), Sentinel won't know that `AccountName` refers to a user. Mapping tells Sentinel how to treat that field—enhancing automation, hunting, and incident triage.

In Microsoft Sentinel, the type of detection rule you choose directly shapes how quickly you detect threats, how accurate the alerts are, and how much operational effort is required to maintain them. Near-real-time rules are designed for fast-moving scenarios where minutes matter, and scheduled analytics rules provide flexible KQL-based detections that you can tailor to your environment.

Types of detection rules

Microsoft Sentinel supports several types of detection rules, each tailored to different operational needs within SOC. The most common are **scheduled** rules. These run custom KQL queries on a recurring schedule, typically every few minutes or hours, against a rolling window of ingested data. These rules are best suited for known threat patterns and deterministic logic. They provide fine-tuned control over alert thresholds, suppression logic, event grouping, and more, making them foundational in most SOC detection strategies. For example, scheduled rules detect a sequence of failed login attempts followed by a success or identify the execution of sensitive administrative commands.

Near-real-time (**NRT**) rules are a specialized subset of scheduled rules optimized for speed and responsiveness. They can run every minute on the latest data, allowing near-real-time detection. Because of this tight interval, NRT rules come with limitations. You cannot use joins, unions, or complex logic functions, but the trade-off is well worth it for detecting high-risk behaviors such as privilege escalation or unexpected changes to critical roles. They are particularly valuable when the SOC needs to act within seconds of an event occurring.

Alongside these, **anomaly-based** rules leverage ML models trained within Sentinel to identify deviations from normal behavior. Rather than looking for known bad actions, anomaly rules detect when a user or device does something statistically unusual, such as accessing far more data than usual, authenticating from a new region, or behaving differently compared to their historical baseline. These rules are especially useful for catching unknown or novel threats, and they complement the more deterministic scheduled and NRT rules by highlighting outliers that might otherwise go unnoticed.

Together, scheduled, NRT, and anomaly-based rules form a layered detection strategy. Scheduled rules bring precision, NRT rules bring speed, and anomaly rules bring adaptability, ensuring that the SOC can respond to both known threats and emerging behaviors in real time.



What happened to the fusion rule?

If you're wondering about the **fusion analytics** rule, there's been a key change. When you onboard Microsoft Sentinel to the unified SecOps platform, the fusion rule is automatically disabled. That's because the unified platform now relies on Microsoft Defender XDR's built-in incident creation and correlation capabilities. These enhanced correlation features replace the legacy Fusion engine, providing a more integrated and intelligent approach to incident management across Microsoft security products.

Entity mapping

By mapping entities, Microsoft Sentinel classifies and enriches alert data for use in incidents and investigations. This helps correlate activities across multiple rules and data sources and finds meaningful context in workbooks, hunting queries, and incident timelines. This results in transforming raw telemetry into structured, recognizable entities, significantly improving the analyst experience.

Let's now look at an example where entity mapping is done.

Example use case: Files Copied to USB Drives

Let's jump back to our USB scenario rule, as touched on earlier. In a rule designed to detect file transfers to USB storage, you can map the following entities, as shown in *Figure 8.32*:

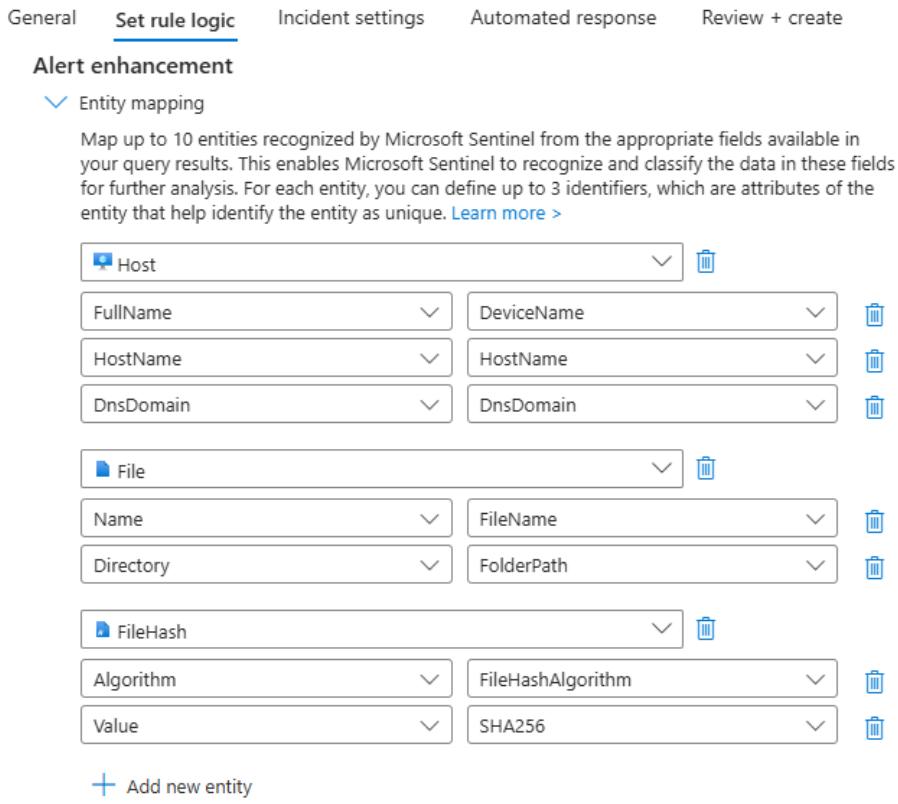


Figure 8.32: Entity mapping in analytic rule

- **Host**: Identifies the endpoint or server where the activity occurred. The entries in this example are **DeviceName**, **HostName**, and **DnsDomain**.
- **File**: Captures which file was copied and its original directory location. Here, they are **FileName** and **FolderPath**.
- **FileHash**: Associates a cryptographic fingerprint with the file to track or correlate its presence across multiple events. The entries here are **FileHashAlgorithm** and **SHA256**.

Each of these entities can be mapped using up to three unique identifiers, enabling Sentinel to distinguish them accurately within queries and incidents.

Once entity mapping is configured, its benefits show up across several areas of the SOC workflow:

- Included in alert metadata and visualized in investigation graphs
- Reflected in entity behavior profiles via UEBA
- Used in correlation logic to stitch related activities into a single incident

In short, entity mapping enhances detection quality and streamlines triage by transforming low-level data into actionable intelligence. This empowers SOC analysts to identify patterns, reduce noise, and respond with greater precision.

Configuring query scheduling in an analytics rule is critical for balancing performance, alert fidelity, and incident response efficiency within the SOC.

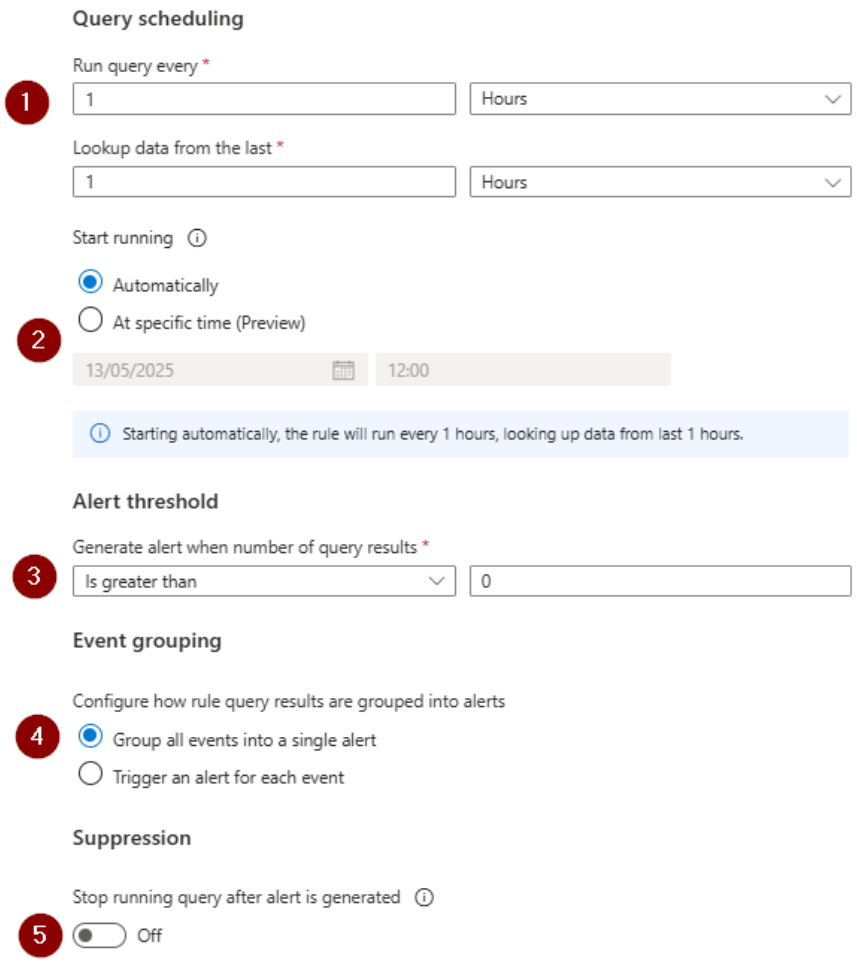


Figure 8.33: Query scheduling settings

In the example shown, the rule runs every hour and queries the past hour's data, a 1:1 ratio between frequency and lookup window. This setup ensures complete coverage with no data overlap or gaps, making it ideal for time-sensitive detections where precision is key. The rule is configured to start automatically upon creation, streamlining deployment without requiring a specific go-live time. This is especially useful in agile SOCs where detection rules must be iterated and deployed quickly.

Alert threshold is set to trigger when the result count is greater than zero, which means that any match from the query will generate an alert. This is well-suited for high-fidelity detections where even a single event could indicate compromise or suspicious behavior. For **Event grouping**, the rule consolidates all matching events into a single alert. This reduces noise and simplifies triage, especially important in high-volume environments where analysts need context-rich alerts, not alert fatigue. **Suppression** is disabled, allowing the rule to continue generating alerts even after a prior match. This ensures ongoing visibility, particularly in cases where repeated activity may suggest lateral movement or persistence attempts.

Optimizing incident handling in Microsoft Sentinel

One of the most impactful factors influencing day-to-day operations in an SOC is how incidents are generated and managed. The way alerts are grouped, when incidents are created, and how recurring activity is treated can either streamline investigations or overwhelm analysts with noise. *Figure 8.34* shows the settings to enable to optimize incident handling.

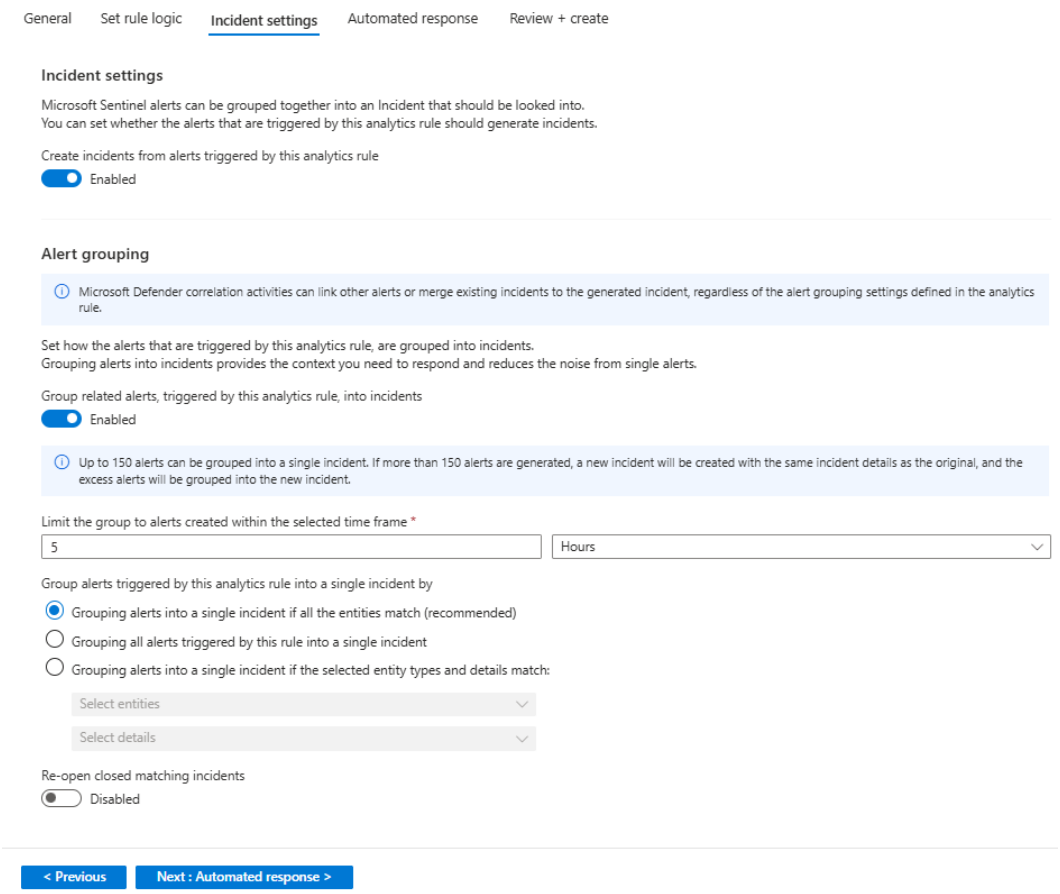


Figure 8.34: Incident settings

By default, Microsoft Sentinel can automatically create incidents for alerts triggered by a rule. While this helps ensure that alerts aren't missed or lost, enabling incident creation without proper tuning can quickly lead to alert fatigue. For high-volume environments or those prone to false positives, this may reduce efficiency and slow down response times.

To mitigate this, Sentinel supports **Alert grouping**, which allows multiple related alerts to be bundled into a single incident if they meet defined conditions. This is particularly useful in reducing clutter. For example, if a file is copied to a USB drive multiple times on the same host within a few hours, grouping prevents the SOC from receiving 10 separate, but nearly identical, incidents.

But grouping isn't just about cleanup; it shapes how investigations unfold. Sentinel's built-in correlation logic can link alerts across different rules and data sources, such as Defender for Endpoint and Defender for Identity, presenting a unified attack narrative instead of isolated events. This saves analysts time and enables faster root cause analysis.

To prevent overly large incidents, Sentinel includes default safeguards or built-in safeguards:

- **Maximum of 150 alerts per incident**: Once this limit is reached, a new incident is created
- **Time-based grouping window** (e.g., 5 hours): This defines how far apart alerts can be and still be grouped together

These controls help create predictable, manageable incident sizes, aiding triage and prioritization.

Sentinel also offers flexible options to refine how alerts are grouped:

- **Attribute-based grouping**: Only group alerts if specific fields, such as **Host**, **filename**, and **User**, match. This ensures that all alerts in an incident share meaningful context and aren't mistakenly lumped together.
- **Broad grouping**: Group alerts even without entity matches. This is useful for detecting wider campaigns or handling simple logic that doesn't require tight entity correlation.

You can also configure whether closed incidents should reopen if a matching alert appears again. While often disabled to avoid churn, this setting can be valuable in tracking persistent threats like beaconing malware or repeated brute-force attempts, ensuring continuity without losing historical insight.

Automating response within Analytics Rules

In the **Automated response** section of the Microsoft Sentinel **Analytics Rules** wizard, SOC teams can move beyond detection and directly link alerts to action. This isn't just about generating alerts; it's about translating detections into expected outcomes.

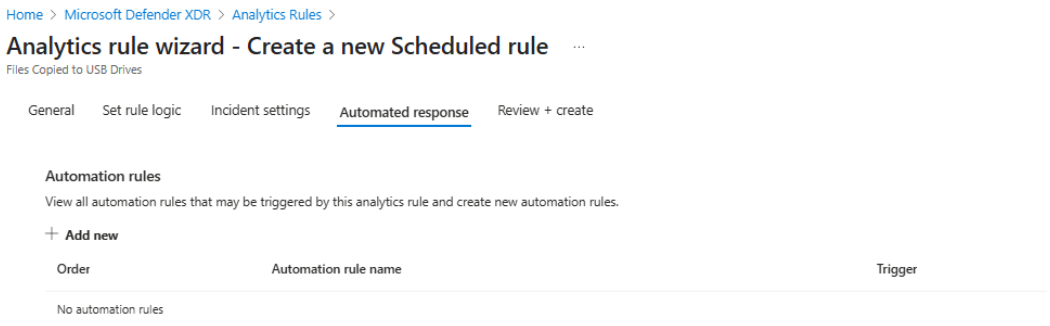


Figure 8.35: Analytic Rules Automated response

In this example, no automation rules have been configured yet. However, the interface provides a streamlined path to define automated actions that should follow a triggered alert.

Automation rules: Bridging detection and response

Automation rules connect alert generation and incident response. Once a rule is triggered, automation enriches incidents with threat intelligence, tags entities for investigation context, assigns ownership to analysts or teams, and launches playbooks that integrate with tools such as ServiceNow, Microsoft Teams, or Defender for Endpoint. This allows SOC teams to encode their response procedures directly into Sentinel, ensuring faster, more consistent triage and a significant reduction in time to resolution.

By embedding automation directly within the analytics rule creation flow, Sentinel promotes a scalable approach to incident handling. Whether the objective is to auto-close low-value alerts, escalate high-severity detections, or pull additional context from logs or threat feeds, automation rules enable an immediate, structured response. *Figure 8.36* shows the creation of a new automation rule.

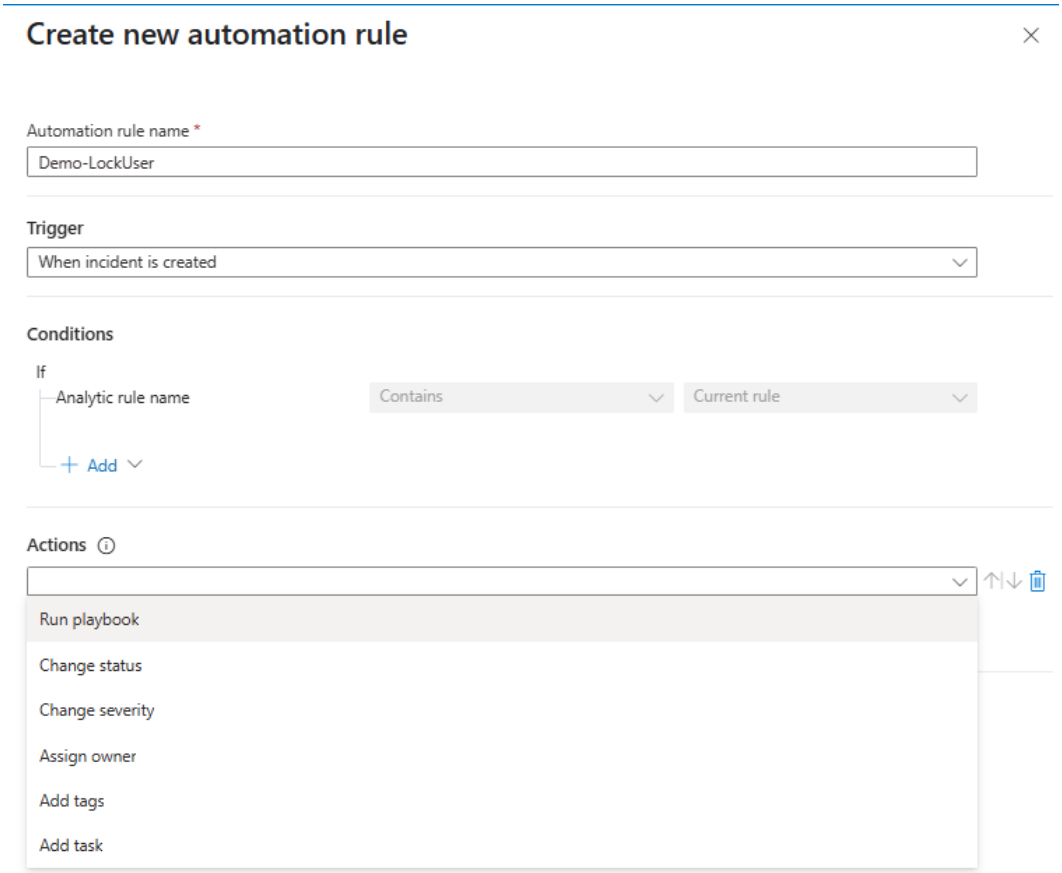


Figure 8.36: Creation of a new automation rule

In this example, the automation rule called **Demo-LockUser** is set to trigger as soon as an incident is generated. That immediate timing is intentional because it ensures the SOC doesn't lose valuable seconds between detection and response. For high-impact threats, acting quickly can make a real difference in limiting damage. The condition is scoped specifically to the analytics rule that triggered the incident. This keeps the automation tightly aligned with the detection logic and ensures that the response is relevant to the threat at hand.

The real value comes through in the **Actions** menu. Sentinel goes beyond basic alert enrichment. You can launch a playbook that runs a Logic Apps workflow. That workflow might disable a user in Entra ID, isolate a device, or send a notification to stakeholders through Microsoft Teams. You can also update the incident severity, assign it to the right analyst, or add investigation tasks to guide the response process.

What makes this setup especially effective is how modular it is. You can chain multiple actions to a single trigger, building flexible, policy-driven workflows that execute automatically. Whether you are enriching incidents, applying escalation procedures, or initiating complex incident response steps, these automation rules allow Sentinel to operate as a true extension of the SOC. It helps ensure consistent, fast, and reliable responses without waiting for manual intervention.

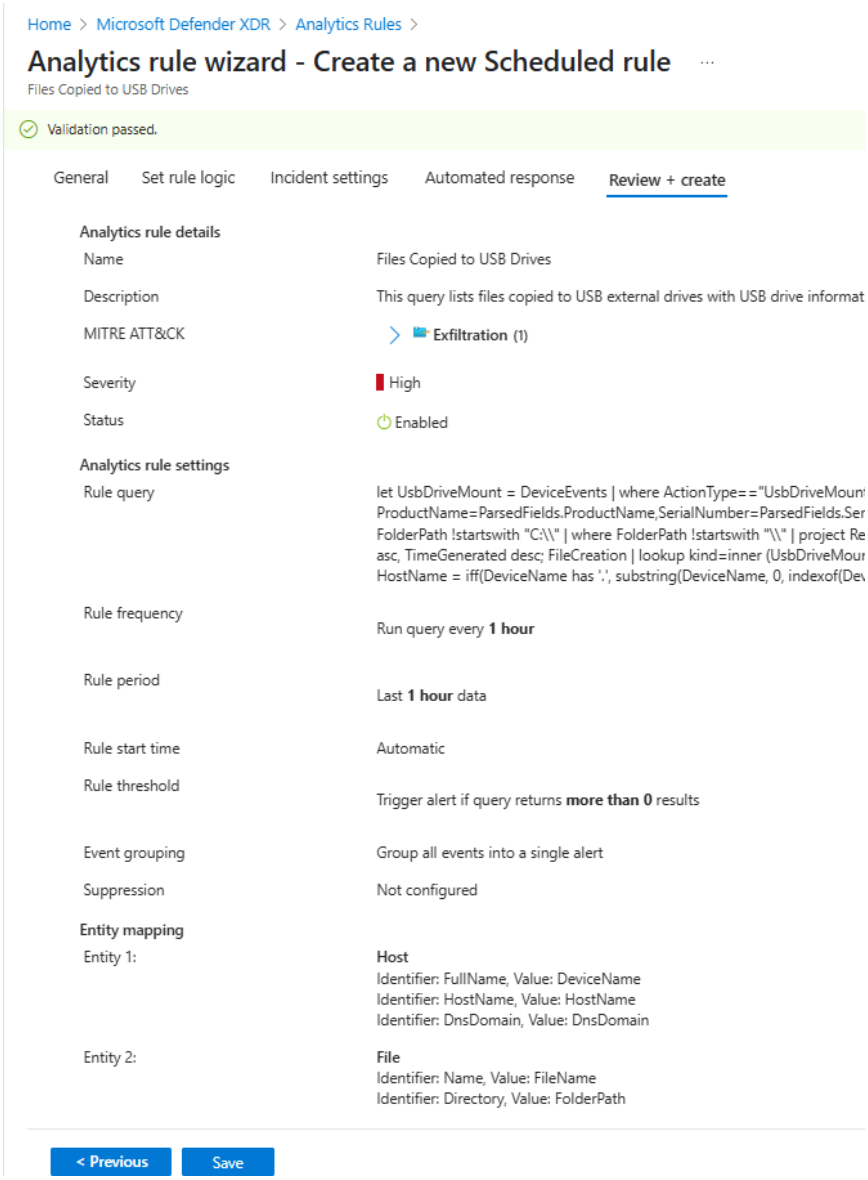


Figure 8.37: Analytics rule wizard – new schedule rule creation

See the overview screen ready to create a new rule. And then it becomes live.

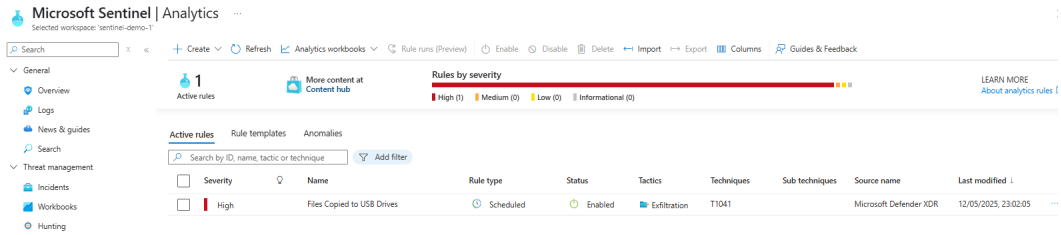


Figure 8.38: Analytics rule – severity settings

Optimizing detection rules in Microsoft Sentinel is not just about catching threats. It is about doing so efficiently, accurately, and at scale. Rule engineering sits at the core of a mature SOC and is often one of the most skilled, strategic, and high-impact roles in the team. It is far from a one-time setup task. Building effective detection rules requires a deep understanding of the environment, data sources, threat landscape, and operational goals, along with the ability to continuously refine and validate logic over time.

At its foundation, rule optimization begins with visibility. You cannot detect what you cannot see. That means aligning rules with the right data sources is essential. Sentinel supports a broad set of connectors, each bringing its own tables, schemas, and event types. For instance, a rule targeting sign-in anomalies in Entra ID will rely on the `SigninLogs` table, while a rule for detecting lateral movement on endpoints will lean on `DeviceNetworkEvents` from Defender for Endpoint. Understanding the specifics, such as table structure, field naming, and ingestion latency, directly impacts how rules are written and how well they perform.

From there, strong rule authorship goes beyond static logic. Rigid *if this, then that* rules are prone to noise and false positives. High-quality rules are dynamic and use KQL variables for reusable conditions, apply joins across tables for deeper context, and incorporate thresholds that reflect baselines or entity-specific behavior. For example, instead of alerting on *more than 20 failed logins*, a smarter rule would compare against that user's normal login activity or adjust for time of day. This improves fidelity while reducing alert fatigue.

Threat intelligence adds another layer. Detection rules should not operate in isolation. They need to integrate with threat feeds, whether from Microsoft or custom sources. Matching IPs, hashes, or domains to known threat indicators shifts rules from simple policy checks to proactive threat detection. However, this brings complexity. Threat intelligence varies in quality, has an expiration, and requires constant curation. Engineers must validate both the intelligence source and how it is used, whether in a join, a lookup, or part of an enrichment process.

Most critically, rule engineering is not a *set it and forget it* job. Rules must be tested, versioned, and continuously validated using red team simulations, threat emulation, and production telemetry. Leading SOCs treat detection as code, building pipelines that support peer review, regression testing, and documentation. A good rule is not just functional in KQL. It is backed by context, tuned over time, and justified by the purpose it serves.

This is why rule creation should never be left solely to junior staff or handed off to automation. It requires technical expertise, threat understanding, environmental awareness, and strategic thinking. In many ways, the effectiveness of a SOC is defined by the strength of its detections, and detection engineering is where that strength is built.

To learn more about KQL to allow you to better engineer rules, see the useful resources below.

`kqlsearch.com`	**KQL Search Engine**
MustLearnKQL blog series (`https://github.com/rod-trent/MustLearnKQL`)	Must Learn KQL book!
Kusto Insights Newsletter (`https://kustoinsights.substack.com/`)	Kusto Insights newsletter
The Definitive Guide to KQL (`https://www.amazon.com/Definitive-Guide-KQL-Operations-Defending/dp/0138293384`)	Using Kusto Query Language for Operations, Defending, and Threat Hunting
Kusto Query Internals (`https://identityandsecuritydotcom.wordpress.com/wp-content/uploads/2020/05/kql_internals_sentinel.pdf`)	Hunting TTPs with Azure Sentinel
Microsoft Sentinel Analytics Rules (`https://analyticsrules.exchange/`)	Microsoft Sentinel Analytics Rules
KQL Internals (`https://identityandsecuritydotcom.wordpress.com/wp-content/uploads/2020/05/kql_internals_sentinel.pdf`)	Kusto Query Internals: Hunting TTPs with Azure Sentinel
Advanced Hunting Expert Training (`https://learn.microsoft.com/en-us/defender-xdr/advanced-hunting-expert-training`)	Advanced Hunting Expert Training

Table 8.11: Resources for more information on KQL

Introduction to Summary rules

Summary rules, currently in preview, represent a strategic shift in how security teams can pre-process and distill telemetry at scale. This feature enables users to define recurring, scheduled KQL queries that run quietly in the background. These rules are purpose-built to extract specific insights from large datasets without overwhelming the active analytics layer. The focus is not on detecting anomalies in real time, but on producing structured, time-based aggregations that support broader detection and hunting objectives. *Figure 8.39* shows the interface of **Summary rules** in Sentinel.

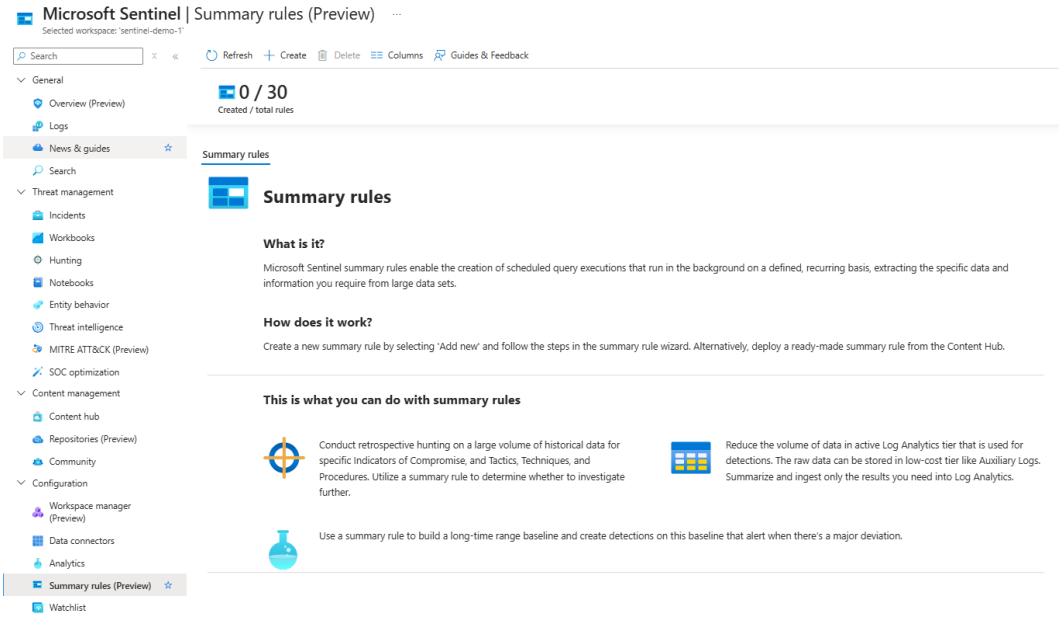


Figure 8.39: Microsoft Sentinel Summary rules

From the interface, you can create summary rules using a dedicated wizard or select from curated options available in the content hub. Operationally, this unlocks capabilities such as retrospective hunting across long-term data, allowing teams to trace indicators of compromise or adversary techniques over weeks or months of telemetry. These rules also help reduce cost and improve performance by summarizing data before it is pushed into the more expensive LAWS. Instead of retaining all raw telemetry at full price, summarized results can be stored in **Auxiliary Logs**, preserving key insights while easing storage and query load.

Summary rules also enable security teams to establish behavioral baselines, which can then inform more precise alerting. Instead of triggering on every event, alerts can be tuned to fire only when meaningful deviations occur, reflecting real shifts in user or system behavior. This approach bridges the gap between cost-effective log management and intelligence-driven security operations.

Creating a summary rule

When creating a summary rule in Microsoft Sentinel, start by defining a clear name and description. This is essential not only for clarity but also for long-term manageability, especially in environments where dozens of rules may be active for various use cases such as threat hunting, behavior baselining, or compliance reporting. *Figure 8.40* shows the **Summary rules wizard** page to create the same.

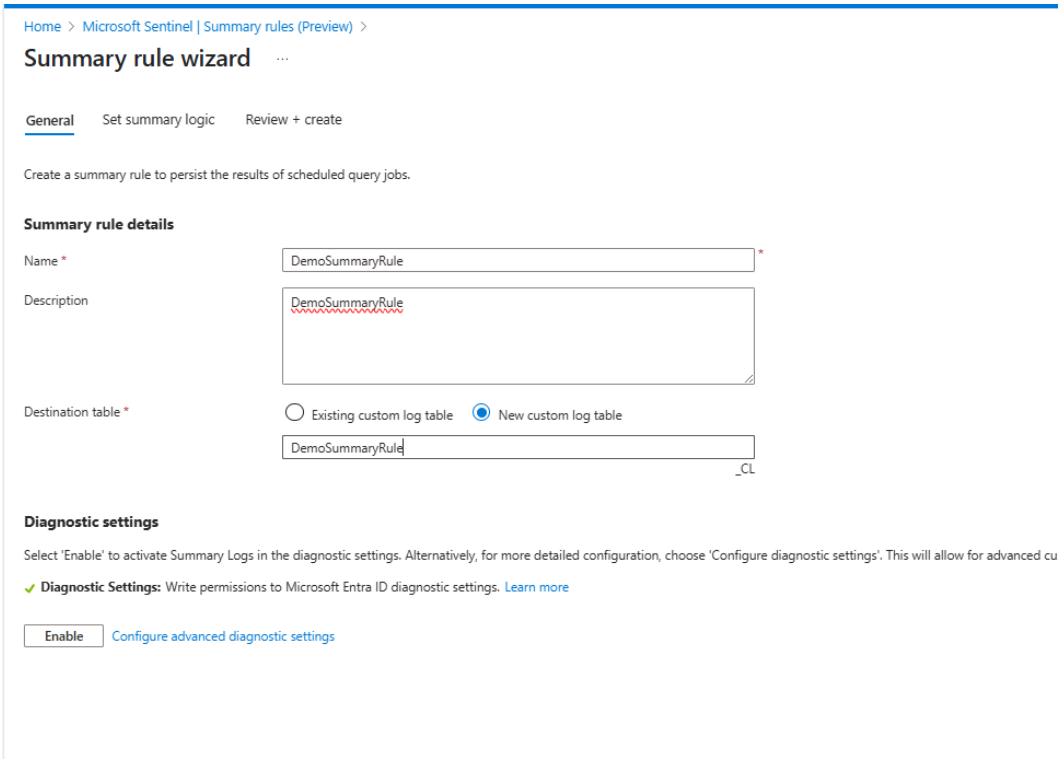


Figure 8.40: Microsoft Sentinel Summary rules

The most important decision to take here is to select the destination table. This is where the results of your summary query will be stored. You can choose to write to an existing custom log table or create a new one. Naming conventions typically follow Kusto standards, such as using the `_CL` suffix. This step transforms summary data into a reusable asset that other queries, workbooks, or analytics rules can use. Instead of scanning raw logs repeatedly, you enable efficient access to preprocessed, structured insight.

You also have the option to configure diagnostic settings. This allows integration with Microsoft Entra ID logs or other diagnostic pipelines, ensuring that the execution of your summary rule is auditable and governed according to enterprise monitoring and compliance requirements.

Practical tips for using summary rules

Consider the following tips before using summary rules:

- **Preserve insights efficiently**: Summary rules store aggregated data, which is much more efficient than keeping full raw logs. This makes them ideal for long-term retention of high-value insights at a lower cost.
- **Reduce resource usage**: If you have queries that are compute-intensive or frequently used, consider converting them into summary rules. Instead of scanning entire datasets each time, you can write results to a lightweight custom log table and query that output directly. This saves time and processing resources.
- **Build automation on top**: Summary rule output can feed directly into other analytics rules, including NRT rules. This means you can trigger alerts the moment new data is added to your summary table, keeping your team informed without delay.
- **Work across log tiers**: You can extract valuable signals from cost-effective log tiers such as **Auxiliary** or **Basic** logs using summary rules, then forward the refined results into higher-tier analytics tables where full KQL capabilities are available.
- **Summarize high-volume tables**: For heavy tables such as `MicrosoftGraphActivityLogs`, create summaries by user, device, or application and store them in a custom summary table. This allows you to retain meaningful data while letting raw logs expire quickly, helping you maintain both visibility and cost control.

Now that we've covered the purpose and trade-offs of Sentinel's detection rule types, the next step is improving signal quality by adding business context. This is where watchlists become valuable: they let you enrich detections with the details your organization cares about, such as VIP users, critical servers, high-risk vendors, approved admin workstations, sanctioned geographies, or known maintenance windows, so rules can make smarter decisions.

Deploying watchlists

With watchlists deployed in Microsoft Sentinel, you gain a flexible way to bring external intelligence or internal context directly into your detection logic. This sets the stage for more targeted and relevant alerts.

This view of the **Watchlist** blade in Microsoft Sentinel highlights one of the most versatile tools for embedding contextual intelligence into the SOC's detection and response workflows. What you see in *Figure 8.41* is not just a collection of static reference files. These are curated, dynamic lookup tables that enable analytics rules, hunting queries, and automation playbooks to adapt based on real-world organizational context.

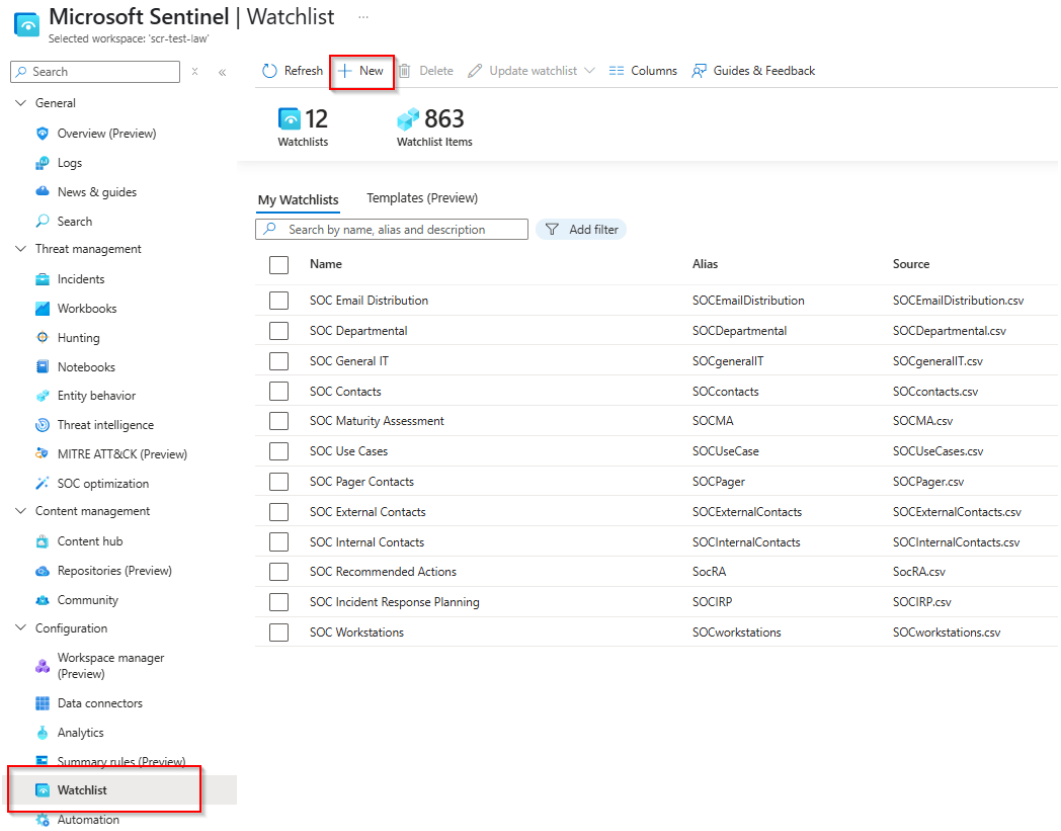


Figure 8.41: Microsoft Sentinel Watchlist

In this environment, the SOC has developed a broad set of watchlists, including **SOC Email Distribution** lists, internal and external contacts, incident response procedures, and maturity assessments. Each list is backed by a source file in the `.csv` format, typically maintained and

updated by operational teams outside Sentinel. These files are imported regularly to ensure the data remains accurate and relevant. The value of watchlists lies in their role as living knowledge objects. For instance, a detection rule can refer to a **SOC Workstations** list to suppress alerts from known system accounts or prioritize incidents tied to specific use cases or critical personnel.

Instead of hardcoding values into queries, which leads to brittle logic and poor scalability, watchlists allow for abstraction and maintainability. You can update the content of a list without touching the rule itself. This makes rule logic cleaner, more portable, and easier to manage. More importantly, watchlists can be combined with threat intelligence, entity behavior profiles, or enrichment processes to produce more precise and actionable detections.

In more advanced environments, watchlists become operational connectors between security policy, detection strategy, and automated response. They allow SOC teams to embed business knowledge directly into the detection layer. This transforms Sentinel from a general-purpose SIEM system into a truly contextual platform that understands not just what is happening, but why it matters.

You can easily deploy watchlists by following these steps:

1. Prepare your data in a CSV or JSON file. This file should contain the list of entities you want to reference, such as IP addresses, usernames, device identifiers, or other relevant fields.
2. In Microsoft Sentinel, go to **Configuration**, then select **Watchlist**.
3. As shown in *Figure 8.42*, the wizard starts with the **General** tab, where you provide a **Name**, **Description**, and **Alias** value for the watchlist.

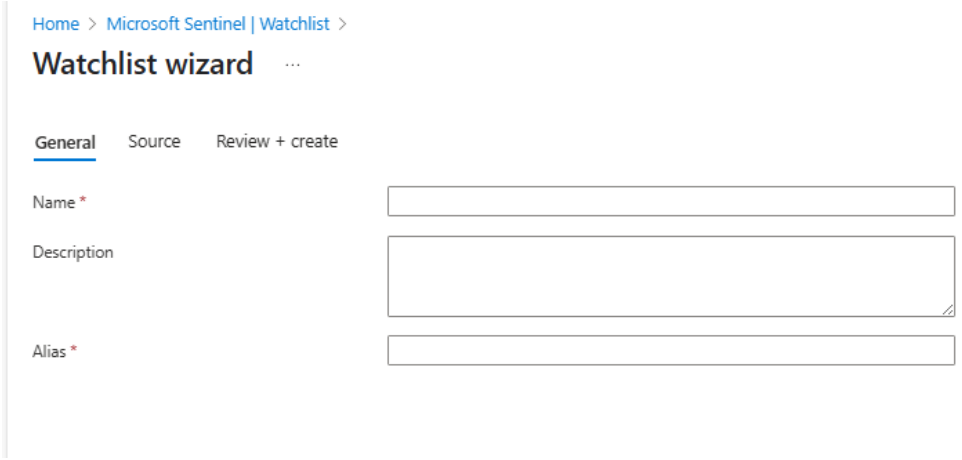


Figure 8.42: Microsoft Sentinel Watchlist settings

These fields serve both administrative and operational purposes. The name helps ensure the watchlist is easily identifiable across the environment. The description gives analysts and engineers quick context about its purpose. The alias acts as the programmatic reference that allows the watchlist to be accessed in KQL queries, making it a critical component for integrating the list into your detection and investigation workflows.

4. Then, navigate to the **Source** tab, where you define the data source that will populate the watchlist. As shown in *Figure 8.43*, you can choose to upload a local file or connect to a source in Azure Storage.

   This provides flexibility to match your organization's operational needs and level of automation.

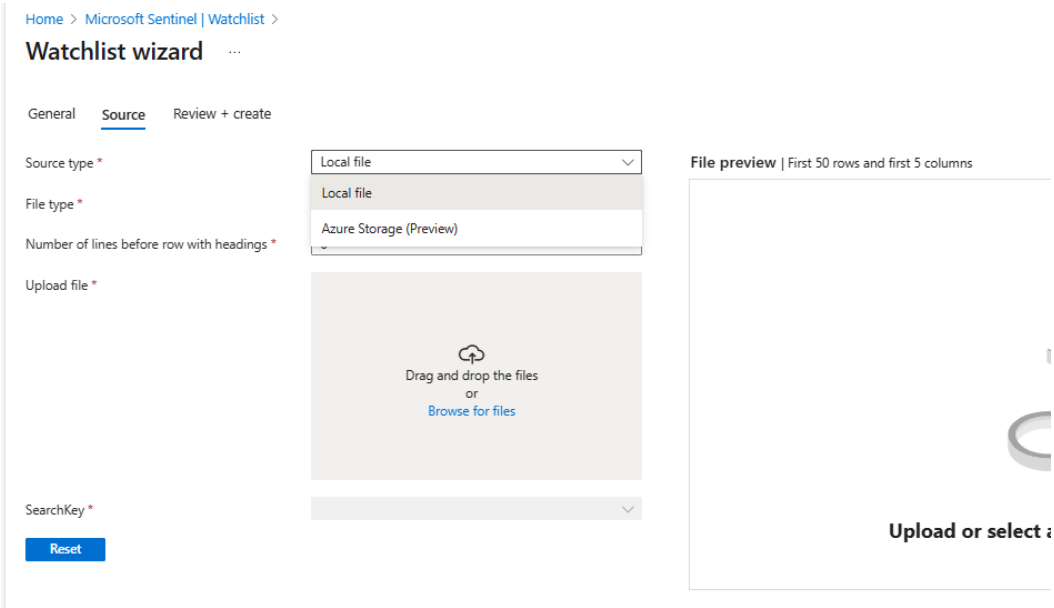


Figure 8.43: Create or update a watchlist

5. Enter the file type and the number of lines before the header row.
6. Set a **SearchKey** value. This key is the column that Sentinel will use in KQL queries to reference or join against the watchlist. It effectively turns the uploaded file into a filterable data object.
7. Once the file is uploaded, you will see a preview pane on the right side of the screen, displaying the first 50 rows and 5 columns. This real-time preview allows you to confirm the file's structure and verify that the correct key column has been selected. It helps catch format issues early and reduces the chance of introducing malformed data into production.

After deployment, monitor watchlist health and update content regularly, either through the portal or by connecting to a dynamic source such as a centralized MSSP location. This keeps the watchlist accurate, aligned with real-world context, and ready for use in active detection and response workflows.

By using watchlists in Microsoft Sentinel, you bring a higher level of precision and relevance into your detection strategy. They enable more intelligent, targeted, and efficient security operations that are grounded in your organization's specific context and priorities. The next step is **Automation**, where detection seamlessly connects to action, turning insights into outcomes without manual intervention.

Introduction to Automation

Microsoft Sentinel's **Automation** feature is designed to take routine, high-volume response tasks out of the SOC team and hand them over to automated logic. Consider *Figure 8.44*, where an automation rule is set.

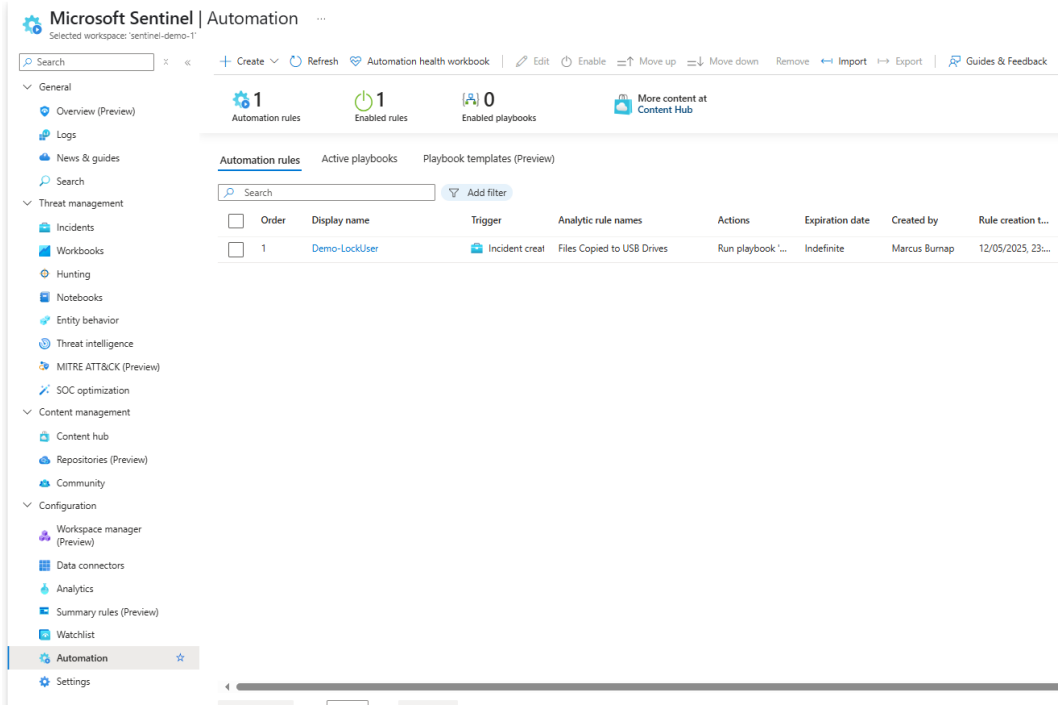


Figure 8.44: Microsoft Sentinel Automation

Here, the automation rule named **Demo-LockUser** is set to be triggered when a specific analytic rule fires, here identified as **Files Copied to USB Drives**. This rule is configured to run a playbook automatically when such an incident is created, helping to enforce policy or respond to risky behavior in near-real time.

Automation rules like this one act as orchestrators for response. They link analytic detections with actions and conditions, essentially saying, "*When X happens, do Y*." In this example, the action is to execute a playbook (likely based on Logic Apps), which can contain steps such as disabling an account, notifying the user, sending an email to a security analyst, or logging the event in a ticketing system. The rule is set with no expiration date, meaning it will continue to run indefinitely until manually changed or removed, and it reflects ownership and auditability by clearly logging who created it and when.

For a SOC team, this interface is where detection meets orchestration. It not only shortens response times but also ensures that reactions are consistent, repeatable, and enforceable, reducing both risk and fatigue. With automation in place, analysts can focus more on hunting and strategic response while Microsoft Sentinel quietly handles the noise and the knowns.

Logic Apps

Logic Apps are the engine behind automation in Microsoft Sentinel. Think of them as the workflow layer that lets you automate anything from sending an email to isolating a machine, all without writing traditional code. Built on Azure's low-code platform, Logic Apps allow you to design sequences of actions and conditions using a visual designer. Each workflow, known as a playbook, connects triggers (such as an incident creation) to actions (such as locking a user or opening a ticket in ServiceNow), often using built-in connectors to Microsoft 365, Teams, Defender, and a long list of third-party services. For SOC teams, Logic Apps are how detection turns into response, instantly and consistently.

With this foundation in place, let's explore a Logic Apps app in action.

Logic Apps use case

In the Microsoft Sentinel **Automation** setting, you can define precise, conditional responses to specific detection events. For instance, consider *Figure 8.45*.

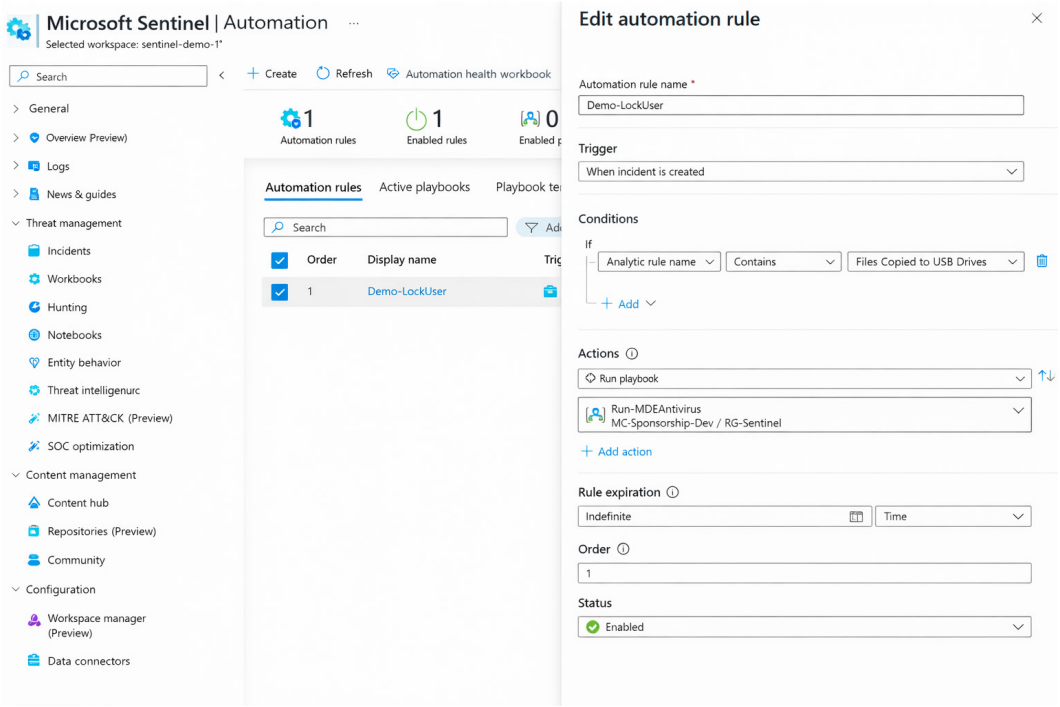


Figure 8.45: Microsoft Sentinel Automation rules settings

The **Demo-LockUser** rule is triggered automatically whenever an incident is created from an analytic rule titled **Files Copied to USB Drives**. Rather than requiring manual triage, the rule immediately kicks off an action. In this case, it launches a Logic Apps app named **Run-MDEAntivirus**. This is where the real automation magic begins.

Click on the playbook (logic app). It will take you into a dedicated design surface within Azure Logic Apps with a visual workflow designer, as shown in *Figure 8.46*, where you can see exactly what actions will be taken.

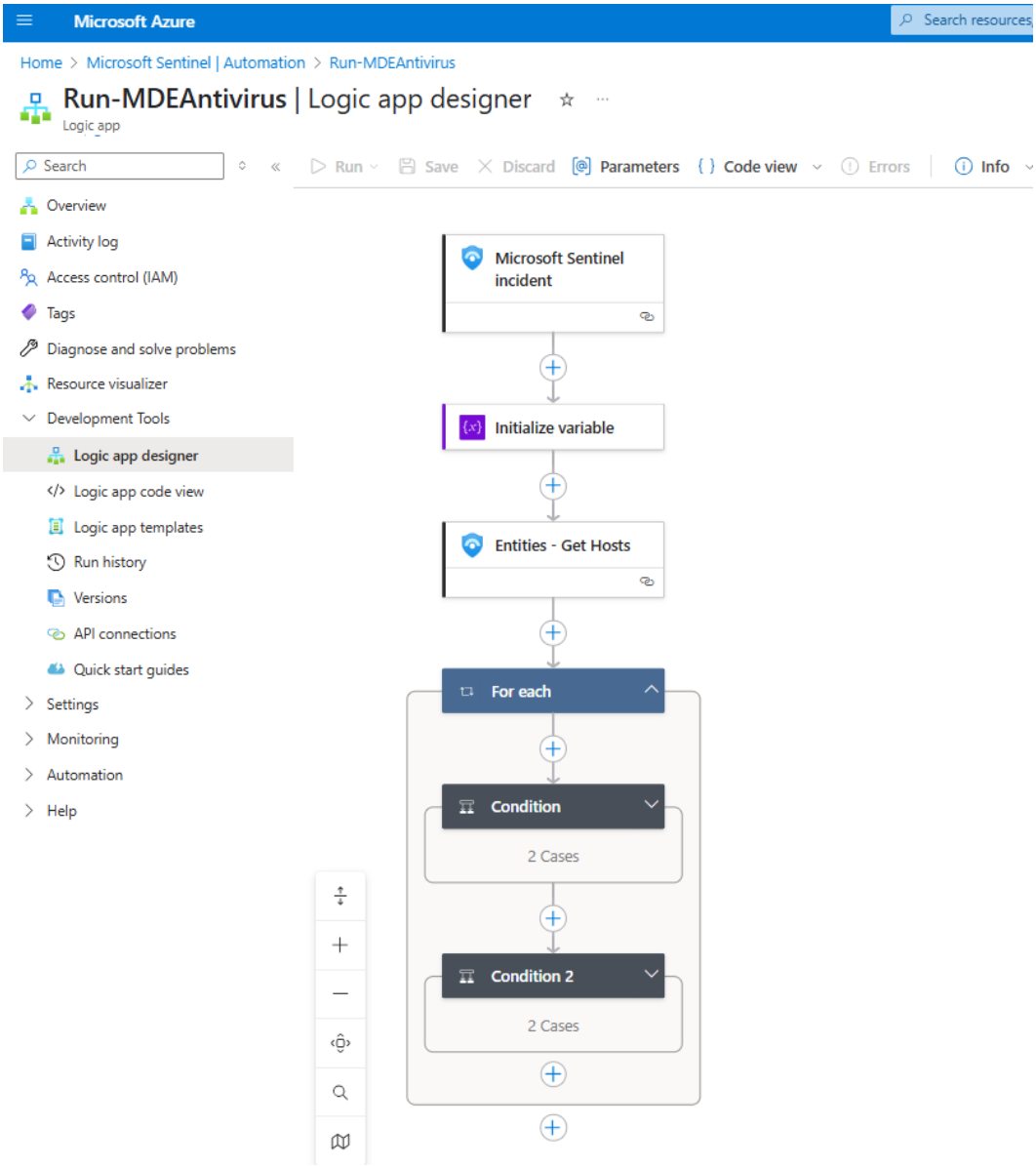


Figure 8.46: Log Apps breakdown

You can see the internal structure of the playbook called **Run-MDEAntivirus**. The flow begins at the top with a **Microsoft Sentinel incident** trigger. This means the Logic Apps app is kicked off as soon as Sentinel creates an incident, allowing immediate response. It then initializes a variable, which is often used to store interim values such as entity names, counters, or flags for later use in the logic.

Next, the playbook uses an **Entities - Get Hosts** step to extract host-related data from the incident, essentially pulling out the machines linked to the detection. Once the host entities are retrieved, the workflow enters a **For each** loop. This means the next steps are repeated for each host identified in the incident, allowing the playbook to scale across multiple affected machines.

Inside that loop, we see two **Condition** blocks. These are logical decision points, essentially, `if/then` statements. Each condition has **2 Cases**, meaning it evaluates whether a certain condition is true or false and then takes different actions depending on the result. For example, one condition might check whether the device is running Windows, and the other might verify whether it belongs to a specific subnet, risk level, or tag.

When configuring an automation rule, you're not limited to a single predefined action; instead, you can choose from a range of Logic Apps that you've already built or imported into your environment. In *Figure 8.47*, you can see several different playbooks ready to deploy, each tied to the **MS-Sponsorship-Dev** resource group.

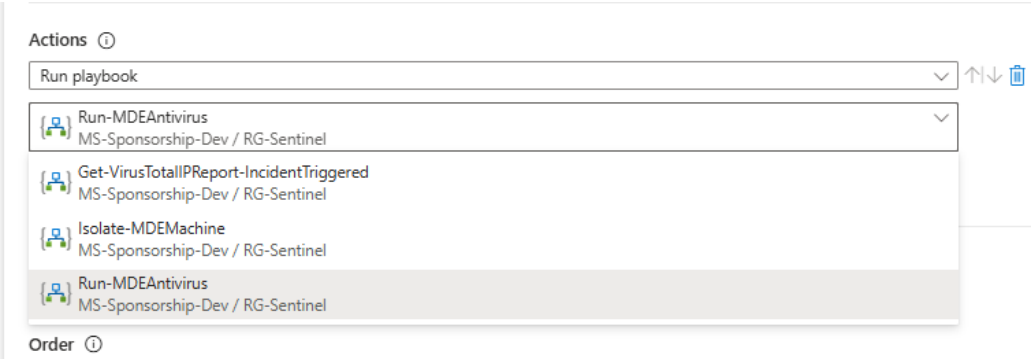


Figure 8.47: Playbooks in Microsoft Sentinel

For instance, **Run-MDEAntivirus** is likely a playbook that triggers a Microsoft Defender for Endpoint antivirus scan on the impacted device. Meanwhile, **Isolate-MDEMachine** probably takes a more aggressive approach, cutting the machine off from the network entirely (except for essential connections to Defender services), a common containment step in high-severity incidents. Then

there's **Get-VirusTotalIPReport-IncidentTriggered**, which likely integrates with VirusTotal to enrich incident data with threat intelligence, automatically querying IPs found in the incident for known malicious activity.

What's important to recognize is that each of these playbooks represents a reusable logic pattern. You're not locked into a one-size-fits-all model; you're building or selecting from a library of responses tailored to different attack scenarios. This modularity lets your SOC respond with the nuance that real-world threats demand. And you can take it even further. These playbooks can be chained together, branched using conditional logic, or triggered based on attributes such as incident severity, entity type, or the specific analytics rule that fired. For organizations aiming to scale their response without increasing headcount, this level of flexibility is a major advantage.

But what if your team doesn't have in-house developers or the time to master the intricacies of building these workflows? That's where Microsoft Security Copilot steps in. Acting as a guided response assistant, Security Copilot helps analysts build logic-based responses without needing to understand the full syntax of KQL or the structure of Logic Apps. It can take a high-level description of what you want to achieve, such as *Isolate the machine and alert the user if suspicious USB activity is detected*, and guide you through crafting a playbook or even generating the workflow directly.

And the future's getting even brighter! Microsoft has already signaled that Sentinel will soon support built-in incident summaries, automatically generated as part of the platform experience. This means Security Copilot will not just help create playbooks, it will also start writing up incidents, summarizing alerts, correlating data, and surfacing recommended actions. So even if you are not a Logic Apps expert or do not have dedicated automation engineers, you're not out of the game. With Copilot's help, you can still build powerful, repeatable, and intelligent responses, transforming your SOC from reactive to proactive and from manual to automated.

We have now deployed the core elements of Microsoft Sentinel.

Introduction to Incidents

At the center of threat management is the **Incidents** feature, the hub where detection meets response. This is the heart of Sentinel's operational workflow. Instead of dealing with isolated alerts, Sentinel correlates them into a single incident, giving you a unified view for triage, investigation, and action. Each incident will have detailed context, timeline data, and direct links to affected users, devices, or other entities, helping your team respond faster and more effectively.

Figure 8.48 shows the **Incidents** section in Microsoft Sentinel.

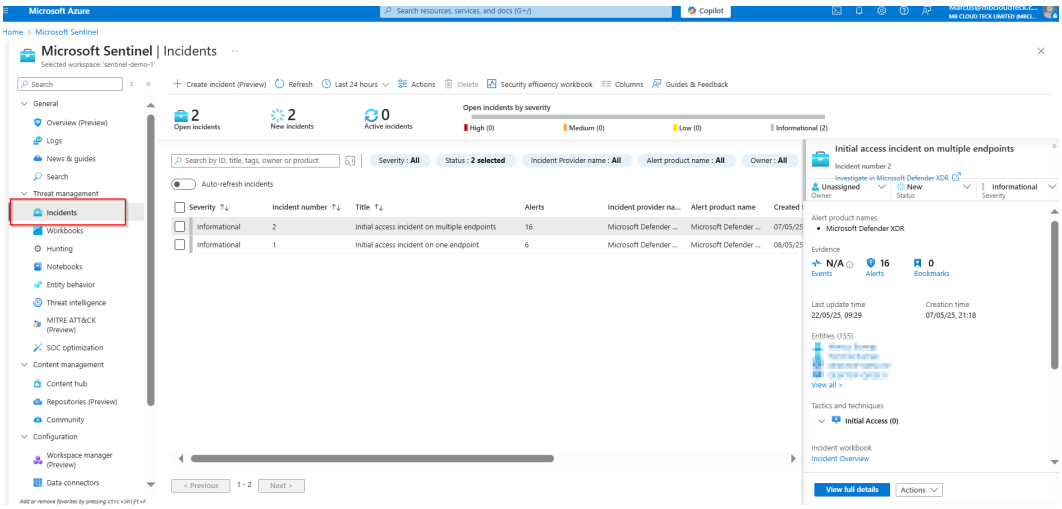


Figure 8.48: Microsoft Sentinel incident queue

At the top center of the screen, you can see a summary that shows the following incidents:

- **2 Open Incidents**: These incidents haven't been resolved or closed yet
- **2 New Incidents**: These incidents have not been triaged, assigned, or updated

The severity breakdown bar just above the incident list shows you a color-coded distribution of incidents by severity (**High**, **Medium**, **Low**, and **Informational**). In this case, both incidents are marked as **Informational**, meaning they are lower priority and don't necessarily require immediate action, but are still relevant.

The middle section presents a grid of open incidents, each showing the following details:

- **Severity**: Indicates the importance of the incident (e.g., **Informational** in this case).
- **Incident number**: A unique identifier used internally in Sentinel.
- **Title**: A summary of what the incident is about. Here, both relate to initial access incidents.
- **Alerts**: The number of alerts grouped into this incident.
- **Provider and Product**: Indicates the source of the detection. Both incidents here came from Microsoft Defender XDR.
- **Created**: The date and time the incident was created.



This table is sortable and filterable, enabling quick triage based on severity, date, provider, or other fields.

Clicking on an incident will open a detail pane on the right. This gives you key metadata and investigation tools without leaving the main page. For the selected incident (**Incident number 2**), you can see the following details:

- **Title**: **Initial access incident on multiple endpoints**, which suggests a broader attack surface
- **Owner**: Currently unassigned, meaning no analyst has taken ownership yet
- **Status**: Still marked as **New**
- **Severity**: **Informational**, suggesting it may be a test event or a low-priority alert grouping

Under **Evidence**, we see the following details:

- 16 alerts are grouped into this incident
- No events or bookmarks are linked yet
- A list of entities involved (users and devices)

The **Tactics and techniques** section maps to MITRE ATT&CK, showing that this falls under **Initial Access**. This refers to techniques adversaries use to gain a foothold in your environment. At the bottom of this pane, there's a link to the incident workbook (**Incident Overview**) and an option to view full details, which would take you to an in-depth investigation view with timelines, visual evidence graphs, and integrated logs.

The **Overview** screen is where triage decisions happen. You can quickly see the scope of incidents, who or what is affected, and decide whether to investigate further or close it. It also helps you prioritize incidents based on the sources, such as Defender XDR, the number of alerts involved, and the mapped MITRE techniques. With the right tuning, this dashboard will help your team detect, assign, investigate, and respond in one workflow.

Let us now look at how we can visualize data within Sentinel.

Introduction to Workbooks

Workbooks are customizable dashboards that let you visualize data from Sentinel and other sources through interactive charts, graphs, and tables. They're great for threat monitoring, executive reporting, or tracking the performance of detection rules, giving teams real-time visibility into their security posture. These visualizations allow you to quickly understand patterns, trends, and anomalies across a wide variety of telemetry.

In *Figure 8.49*, you can see the **Workbooks** blade within Sentinel.

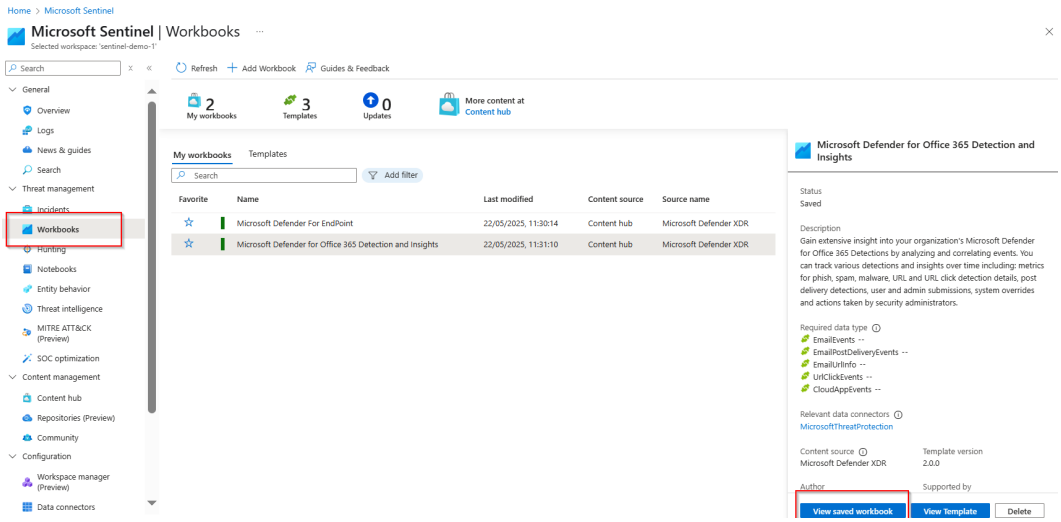


Figure 8.49: Microsoft Sentinel Workbooks

In this section, you can browse existing workbooks, import new ones, or create your own. This particular workspace has two saved workbooks: one for Microsoft Defender for Endpoint and one for Microsoft Defender for Office 365 Detection and Insights. These are sourced from **Content hub**, which provides prebuilt templates curated by Microsoft.

On the right side of the screen, detailed information is shown for the selected workbook, **Microsoft Defender for Office 365 Detection and Insights**. This workbook helps you analyze detection data from Defender for Office 365 by correlating events across signals such as phishing attempts, spam, malware, URL clicks, and email delivery results. It also tracks actions taken by admins and user submissions. Essentially, it gives you a high-level view of Defender's detection posture and the outcomes of protective controls.

Importantly, the workbook is built to operate using specific data types, listed in the **Required data type** section, such as **EmailEvents**, **UrlClickEvents**, and **CloudAppEvents**. This ensures that the visuals and filters within the workbook are accurately populated. If these tables aren't connected to your LAWS, the workbook may display as blank or incomplete. There's also a helpful link to the relevant data connectors (such as **MicrosoftThreatProtection**), so you can easily confirm or enable the required ingestion paths.

Below that, you're given two key options:

- **View saved workbook**: Opens the live, interactive version where you can explore and filter data
- **View Template**: Allows you to duplicate or edit the base template to fit your needs

You can use dropdowns, time range selectors, and filters to refine what you're seeing. For example, you might use a workbook to quickly isolate spam activity across regions, or to identify whether certain users are consistently submitting false positives. You can also extend workbooks by adding new queries, combining metrics from multiple sources, or embedding links to incidents and hunting queries.

The **Microsoft Defender for Office 365 Detection and Insights** workbook shown in *Figure 8.50* is a prime example of how Sentinel workbooks turn raw telemetry into clear, actionable intelligence.

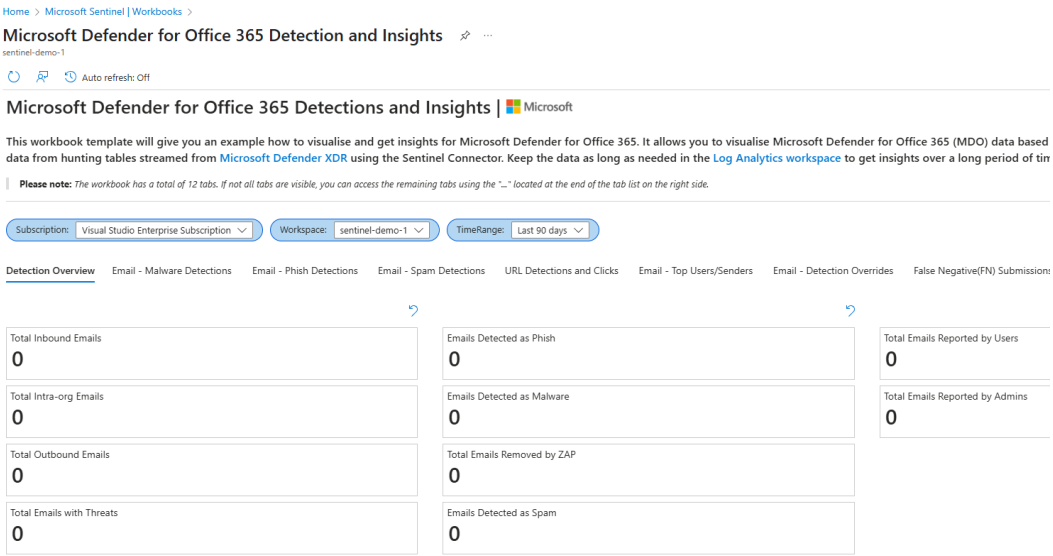


Figure 8.50: Microsoft Defender Workbooks

This workbook consolidates multiple detection streams, such as phishing, malware, spam, and user-submitted threats, into a single, easy-to-navigate dashboard. At the top, you'll see global controls for **Subscription**, **Workspace**, and **TimeRange**, which instantly filter the entire workbook's queries. This is one of the core strengths of Sentinel workbooks. It allows you to interactively change the scope of your analysis without modifying a single line of code.

The workbook is split into interactive tabs, such as **Email - Malware Detections** and **URL Detections and Clicks**, with each tab presenting a focused view into a specific area of Defender for Office 365 telemetry. The **Detection Overview** tab shown here summarizes high-level email traffic and threat activity, totals for inbound, outbound, and intra-org emails, along with threats detected as spam, phishing, or malware. It also includes signals from **Zero-hour Auto Purge (ZAP)** actions and user/admin reports, providing context around human feedback loops within your email protection posture.

Any data ingested into Log Analytics, whether it's from Microsoft tools, third-party integrations, or custom sources, can be queried and visualized in a workbook. You're not limited to security tools either. If the data is there, you can transform it into a chart, graph, table, or KPI widget. The visual components are all powered by KQL, which gives you fine-grained control over filtering, aggregation, correlation, and shaping of the results.

Introduction to Hunting

Microsoft Sentinel provides a dedicated threat hunting workspace designed to help you uncover suspicious activities that may bypass automated detections. Rather than waiting for alerts, threat hunting in Sentinel lets you actively explore raw telemetry, test hypotheses, and trace patterns that indicate adversary behavior.

Figure 8.51 shows the **Hunting** section in Sentinel.

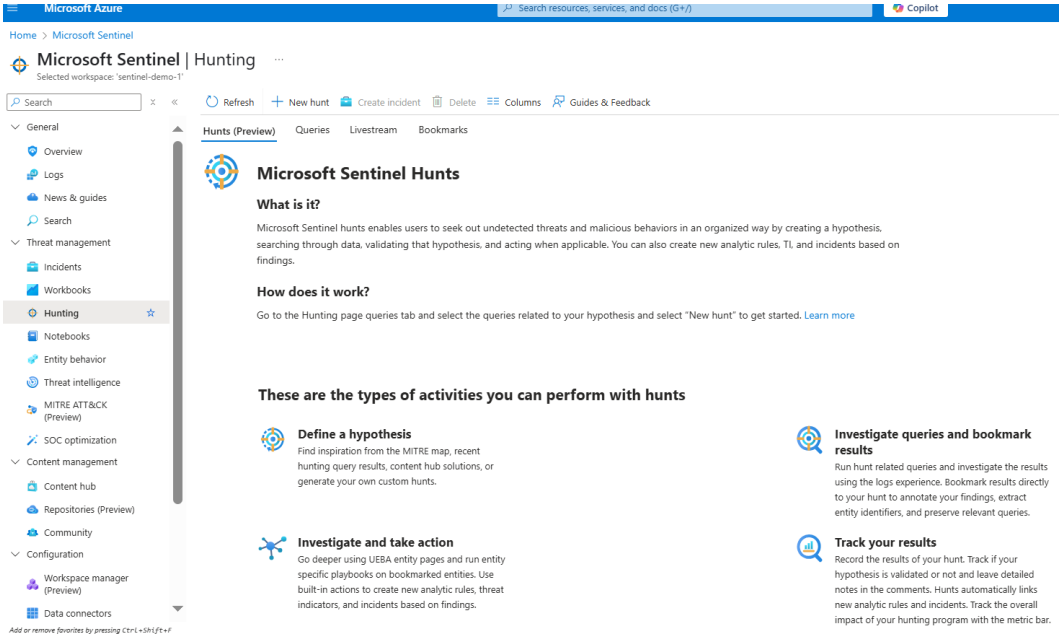


Figure 8.51: Microsoft Sentinel Hunting

This hunting process is organized around hypothesis-driven exploration. You begin by forming a theory, perhaps inspired by known attack tactics, suspicious user behavior, or a new threat intelligence indicator. From there, you use **KQL-based hunting queries** to dig into your connected data sources and uncover supporting evidence. Sentinel gives you tools to bookmark results, pivot between data types, enrich context, and trigger follow-up actions such as creating incidents or generating custom rules.

Hunting in Sentinel is fully integrated with other capabilities in the platform:

- Use MITRE ATT&CK mapping to align your hunts with known tactics and techniques
- Run investigations directly from query results using **Entity Behavior** views
- Leverage bookmarks to track findings and revisit them later in investigations
- Launch playbooks from hunting results for automated response or enrichment
- Document outcomes and track performance through the hunt metrics bar

This structured approach allows you to move seamlessly from raw data to a meaningful response. As you iterate, validated patterns can be converted into new analytic rules or integrated into your SOC workflow.

While Microsoft Sentinel is the ideal place for KQL-driven threat hunting across your SIEM data, the experience becomes even more powerful when it is used alongside Microsoft Defender XDR via the unified SecOps platform. With the unified platform enabled, your Sentinel workspace gains real-time access to Defender's advanced hunting tables, including endpoint, identity, email, and SaaS telemetry. This creates a shared hunting surface across both tools, allowing you to query Defender XDR data directly from Sentinel's interface, correlate SIEM data with XDR activity in one hunting workflow, use M365 Defender's curated threat analytics queries within Sentinel, and share findings, incidents, and hunting results across both platforms.

This integration breaks down silos between detection and investigation, letting your SOC team move from alerts to hypotheses to validated threats across both SIEM and XDR in one connected experience.

Advanced Hunting in Defender XDR

The **Advanced hunting** section in Microsoft Defender XDR unifies Sentinel's hunting schema and data. This integrated approach gives you direct access to Sentinel's tables, queries, and functions from within the Defender portal. It also brings SIEM and XDR together into a single pane of glass for hunting operations.

Figure 8.52 shows the **Advanced hunting** section.

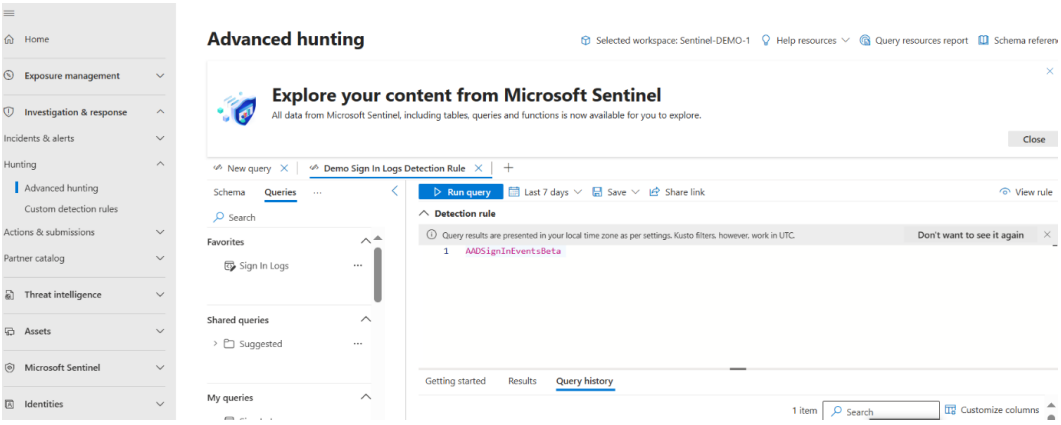


Figure 8.52: Defender XDR Threat hunting section with KQL input section

You can see the currently opened tab titled **Demo Sign In Logs Detection Rule**. The query itself is set to reference the `AADSignInEventsBeta` table, which is a preview schema designed to surface enriched sign-in data from Entra ID (Entra ID). This is one of many schema tables now available thanks to the unification with Sentinel.

The middle section includes the typical hunting interface controls:

- **Run query**: Executes the current KQL-based query
- **Time range selector**: Set to **Last 7 days** to filter the query's scope
- **Save** and **Share link**: Options for collaboration and persistence
- **Detection rule**: Suggests this query may have originated from or is linked to a scheduled detection rule in Sentinel

On the left-hand navigation pane, you'll find the following:

- **Favorites**: Quick access to frequently used queries, such as **Sign In Logs**
- **Shared queries** and **My queries**: Organizational and personal hunting assets for reuse
- **Suggested queries**: Curated by Microsoft based on current threat intelligence or usage patterns

The tab interface and **Query history** pane at the bottom show that this is part of an active investigation. You can switch between multiple open queries or review what's been run previously, which is a valuable feature when pivoting between hypotheses during a hunt.



Started in April 2025, Microsoft Sentinel now ingests all threat intelligence into the newly introduced `ThreatIntelIndicator` and `ThreatIntelObjects` tables. These tables support the STIX 2.1 schema, enabling the ingestion and querying of various threat intelligence objects, including identity, attack patterns, threat actors, and relationships.

Threat intelligence is evidence-based information about adversaries, their tools, and the tactics they use to compromise organizations. It turns raw indicators and observations into context you can act on—what to watch for, what to prioritize, and what to block. Used well, it strengthens detections, accelerates investigations, and improves decision-making by linking activity to known threats. In Sentinel, it also helps you enrich alerts and hunting with trusted indicator and actor context. Let's discover the dashboard.

Introduction to Threat intelligence

The **Threat intelligence** section is used to manage and explore threat intelligence indicators such as IPs, domains, and file hashes. Sentinel can correlate these indicators with your environment to find matches, enabling threat-informed detection and contextual investigations. The section gives security teams a dedicated interface for working with **indicators of compromise (IOCs)**.

These indicators can come from commercial threat feeds, open-source intelligence, or internal sources. Once ingested into Sentinel, these IOCs populate the `ThreatIntelligenceIndicator` table and can be used to enrich logs, trigger analytics rules, or enhance hunting queries.

This is especially useful for Tier 2 and Tier 3 SOC analysts who need to correlate raw security events with known malicious infrastructure, such as IP addresses linked to botnets, domains used in phishing attacks, and file hashes associated with malware.

Figure 8.53 shows the **Threat intelligence** section in Sentinel.

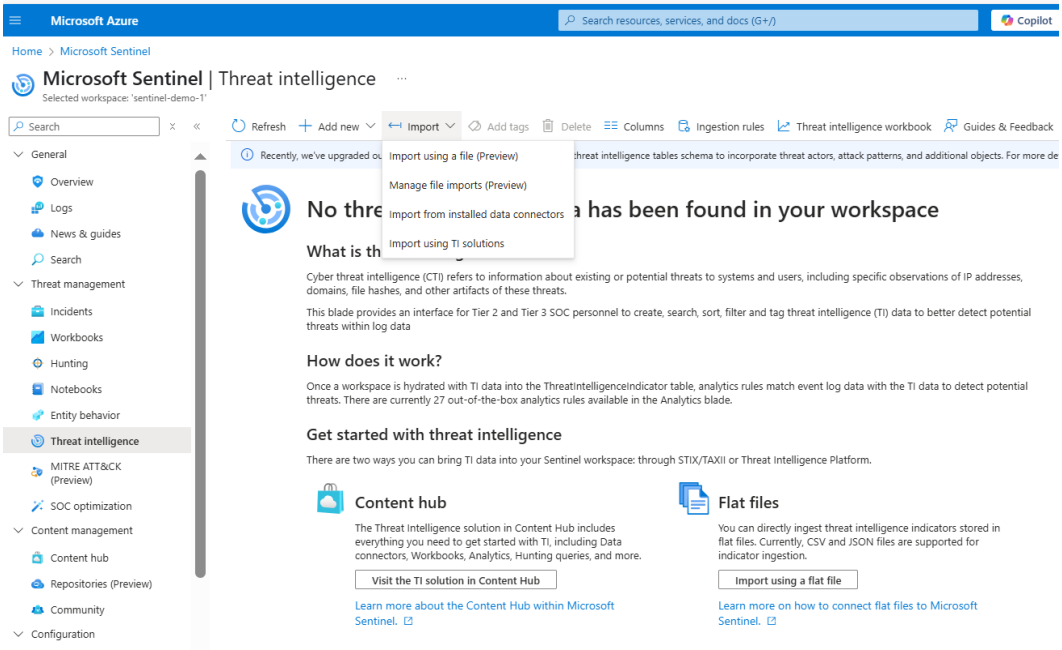


Figure 8.53: Sentinel Threat intelligence blade overview page with the Import section options

At the top of the page, under the **Import** dropdown, there are four main ingestion options:

- **Import using a file (Preview)**: Upload flat files (CSV or JSON) directly containing IOCs. This is ideal for ad hoc updates or internal intelligence sources.
- **Manage file imports (Preview)**: View, edit, or delete previously imported flat files.
- **Import from installed data connectors**: Use built-in connectors from **Content hub** to automatically sync with external platforms (e.g., MISP, Recorded Future, or Anomali).

- **Import using TI solutions**: Integrate via **threat intelligence platforms (TIPs)** such as STIX/TAXII feeds or solutions that publish threat indicators automatically. This method allows scalable, automated ingestion of structured CTI.

Follow these steps to connect using TI solutions:

1. Click **Visit the TI solution in Content Hub**. This will open the **Threat Intelligence** solution package as shown in *Figure 8.54*.

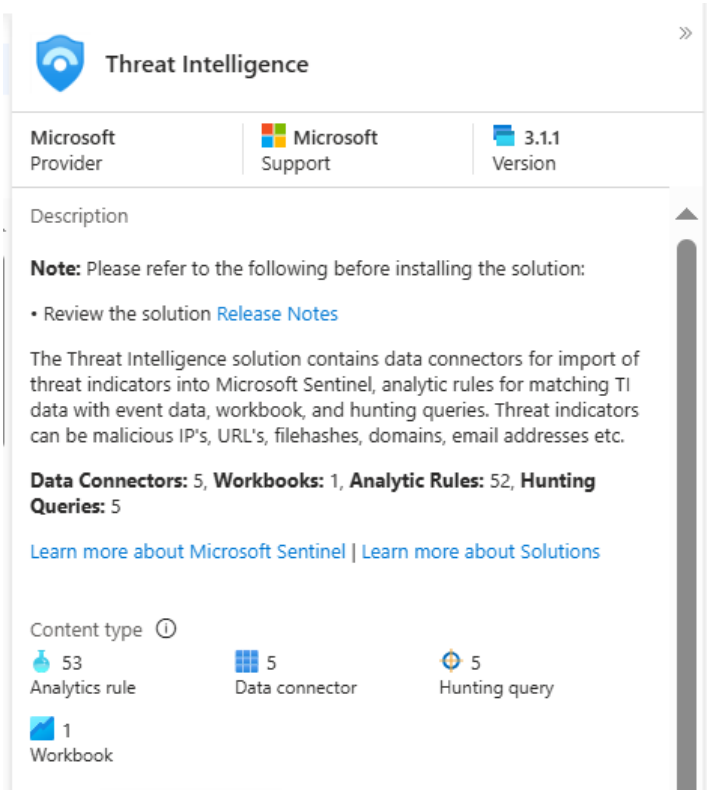


Figure 8.54: Microsoft Threat Intelligence content hub solution connection overview

2. From there, you can deploy the full solution, which includes the following:
    - **Data connectors** for platforms such as TAXII (Structured Threat Information Expression)
    - **Workbooks** for visibility into indicator ingestion and usage
    - **Analytics rules** that trigger when ingested indicators match telemetry in your environment
    - **Prebuilt hunting queries** and dashboards

Once deployed, indicators streamed via STIX/TAXII or partner platforms will automatically populate the `ThreatIntelligenceIndicator` table. They will then be available for matching against logs in real time, enriching incidents or alerts, and strengthening hunting and investigation workflows.

Now that you have threat intelligence in place, the next priority is ensuring that it translates into measurable detection coverage rather than sitting as passive context. The most effective way to do that is to map your analytics to the MITRE ATT&CK framework, which provides a common language for adversary behavior across tactics (the "why") and techniques (the "how"). By aligning Sentinel detections, hunts, and response playbooks to ATT&CK techniques, you can clearly see which parts of the attack lifecycle you can detect reliably, where you are over-alerting on low-value signals, and where you have gaps that an attacker could exploit.

Introduction to the MITRE ATT&CK framework

This view aligns your detections and threats with the MITRE ATT&CK framework, allowing you to track coverage across tactics and techniques. It's a great tool for identifying visibility gaps and organizing your defensive strategy around known adversary behaviors.

As an example, if your threat intelligence indicates active use of credential-dumping tools such as Mimikatz, you can translate that into MITRE ATT&CK coverage by mapping it to **OS Credential Dumping (T1003)** and validating that you have Sentinel detections for the behaviors that support it, such as suspicious access to LSASS Local Security Authority Subsystem Service (LSASS) on endpoints, unexpected privilege escalation to enable memory reads, and follow-on use of stolen credentials for lateral movement, so you are detecting the technique, not just the tool name.

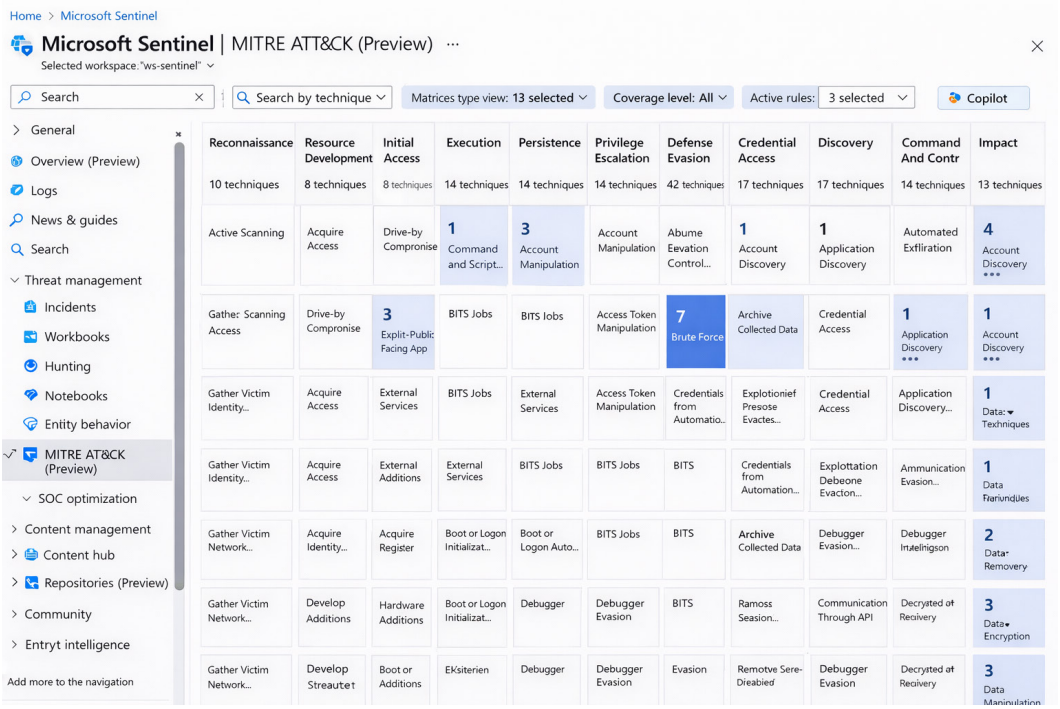


Figure 8.55: MITRE ATT&CK Sentinel blade

Two community tools we would like to highlight to aid you in getting the best coverage when picking or creating rules are the following:

The Azure Attack & Defense Playbook (due to identity being so important). Find the link here: `https://github.com/Cloud-Architekt/AzureAD-Attack-Defense/`. This publication is a collection of various common attack scenarios on Microsoft Entra ID (formerly known as Azure Active Directory) and how they can be mitigated or detected.



All the included scenarios, insights, and comments are based on experiences from the contributors during their attack simulations, hands-on, or real-world scenarios. This provides a rich source of information to understand holistically what you are crafting a new rule for and the context of why you may need a new rule.

The MS Attack Range tool: This is a great way to gauge overall coverage and contextualized testing of your rules.

Find the link here: `oloruntolaallbert/MS-Attack-Range. This is an automated deployment tool that creates instrumented Azure environments with vulnerable systems for simulating attacks and testing Microsoft Sentinel detection capabilities.`

Microsoft Sentinel Attack Range is a practical framework that enables security professionals to deploy a controlled, isolated environment in Azure, generate security events from simulated attacks aligned to MITRE ATT&CK techniques, and then test and validate Microsoft Sentinel analytics rules and detections against realistic attacker behavior. By using it to exercise end-to-end investigations, teams can refine playbooks, reduce false positives, and improve incident response procedures in a safe setting. In conclusion, it provides a repeatable way to prove detection coverage, harden SOC processes, and build confidence that Sentinel will perform as expected when real threats occur.

Summary

This chapter positioned the Microsoft Sentinel deployment plan as a 30-, 60-, and 90-day execution blueprint for taking Sentinel from "installed" to "operational," once the platform foundations and onboarding are already complete. You learned how to build capability in deliberate phases: first, establishing reliable data collection and baseline detections, then optimizing ingestion and visibility while deepening detection, incident handling, and automation, and finally, enabling advanced features such as log tier strategy, threat hunting, UEBA and anomalies, threat intelligence mapped to MITRE ATT&CK, and unified operations with Defender XDR and Security Copilot. The core takeaway is that a structured plan prevents connector sprawl and alert noise, accelerates time-to-value, and delivers a SOC-ready service that can be governed, measured, and continuously improved.

In this chapter, we also tackled one of the latest customer concerns, which is access controls in the Unified Defender Portal.

Since Sentinel is now operating and generating insights, the next challenge is to maintain consistency at scale. The next chapter focuses on how to codify and automate Sentinel deployments across multiple tenants and environments using CI/CD pipelines and infrastructure-as-code practices.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

Note: *Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



9

Microsoft Sentinel Fast-Track Deployment Options and CI/CD Pipelines

This chapter provides a hands-on guide to rapidly deploy Microsoft Sentinel and manage its content using repeatable, scalable CI/CD practices. Whether you're onboarding a new client, setting up a multi-tenant environment, or shifting to infrastructure as code, fast and consistent deployment is essential.

In this chapter, you'll learn about the following topics:

- Deploying and managing Microsoft Sentinel artifacts at scale
- Fast-tracking your IAC Sentinel deployment
- Azure DevOps: project setup

By the end of this chapter, you'll have a working model to automate Sentinel deployment, reduce operational overhead, and maintain consistency across environments, whether you're managing one workspace or hundreds.

Deploying and managing Microsoft Sentinel artifacts at scale

As Microsoft Sentinel deployments mature, the need to standardize, scale, and centrally manage content becomes critical, especially in environments spanning multiple tenants, workspaces, or geographies. This initiative focuses on two core strategies: repositories and workspaces. You can also use API-driven deployments. However, that would justify its own entire book.

Deploying content via repositories

Leveraging the content hub and repositories feature, security teams can publish, version, and distribute curated content packages. These repositories act as centralized blueprints that ensure consistent deployment of detections and dashboards across environments.

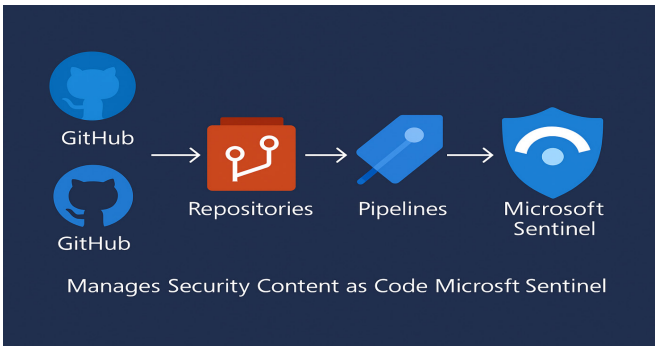


Figure 9.1: Manage security as code in Microsoft Sentinel

This method is ideal for operationalizing threat detection frameworks or aligning all workspaces with internal baselines and compliance standards.

Getting set up to deploy content via repositories

Before you can start managing Microsoft Sentinel content through GitHub or Azure DevOps, there are a few foundational steps you'll need to check off.

First, make sure you've got the right level of access. You'll need to either be an **Owner** of the resource group that holds your Sentinel workspace or combine **User Access Administrator** with **Sentinel Contributor** permissions. That ensures you're authorized to set up the connection.

Next, you'll need access to your source control platform. For GitHub, **Collaborator** access is required, and for Azure DevOps, you'll need to be a **Project Administrator**. Also, make sure GitHub Actions (for GitHub) or Azure Pipelines (for DevOps) are enabled. These are what actually run the deployment automation behind the scenes. Azure DevOps connections must typically live within the same Azure AD tenant as your Sentinel workspace. When connecting across tenants where Azure DevOps is hosted in a home tenant and Sentinel resides in a client tenant, guest user scenarios may require manual URL entry due to interface limitations.

Another critical step is confirming that the content you want to deploy, such as analytics rules, parsers, playbooks, and workbooks, is stored in a supported format. If it's not, Sentinel won't be able to interpret it correctly when deploying.

Connecting your repository

Once you've got your prerequisites squared away, setting up a repository connection is straightforward.

1. In Microsoft Sentinel, head to **Content management | Repositories**, and choose **+Add new**. Give your connection a clear name and a helpful description, which will help you later when you're managing multiple connections.

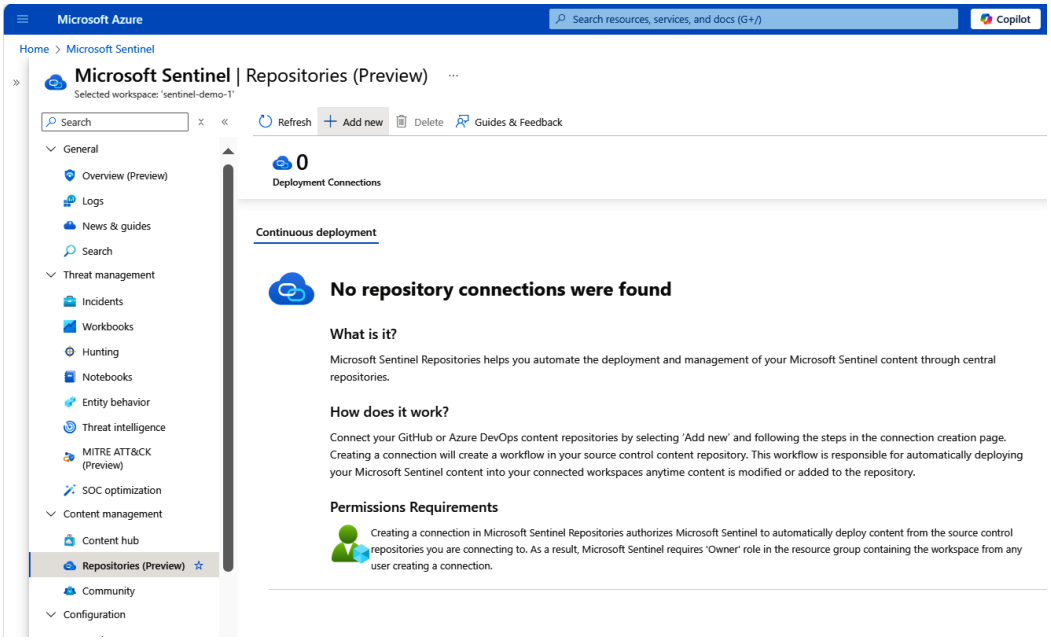


Figure 9.2: New repository blade with no existing connections

2. Then, choose your source control system (**GitHub** or **Azure DevOps**) from the **Source control** dropdown, and select **Authorize**.

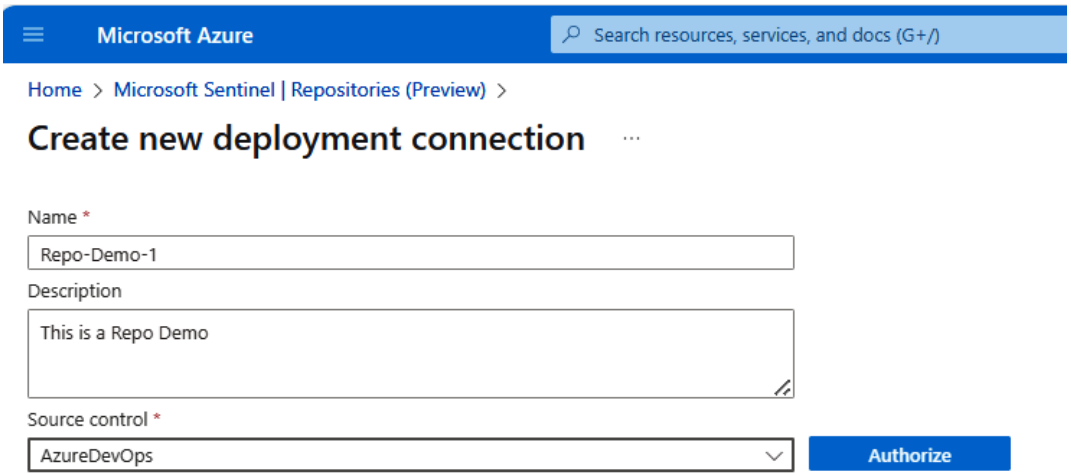


Figure 9.3: Basic repository connection information and authorization to the repository

If you're using Azure DevOps, your browser session will try to authenticate you automatically with your Azure credentials. Just make sure you're authorized to the same tenant; otherwise, it might not show your projects.

Connecting via a client tenant?

Setting up a repository connection for Microsoft Sentinel in a different tenant or separate Sentinel workspace doesn't require much extra effort. Remember that the person initiating the connection by selecting **+Add new** under the **Repositories** tab must have the right level of access to the resource group or subscription where Sentinel lives. Specifically, they should either have the Owner role or a combination of Sentinel Contributor and User Access Administrator.

When it comes to authenticating with Azure DevOps, you'll need to sign in using an account that has the appropriate access in the tenant where the DevOps environment is hosted. This authentication step only needs to be completed once per Sentinel workspace.



If your user is already signed in to Azure DevOps, you may need to remove previous authentications to get the Authentications box to appear and allow you to sign in with the home DevOps account. To do so, visit this link: `https://aex.dev.azure.com`.

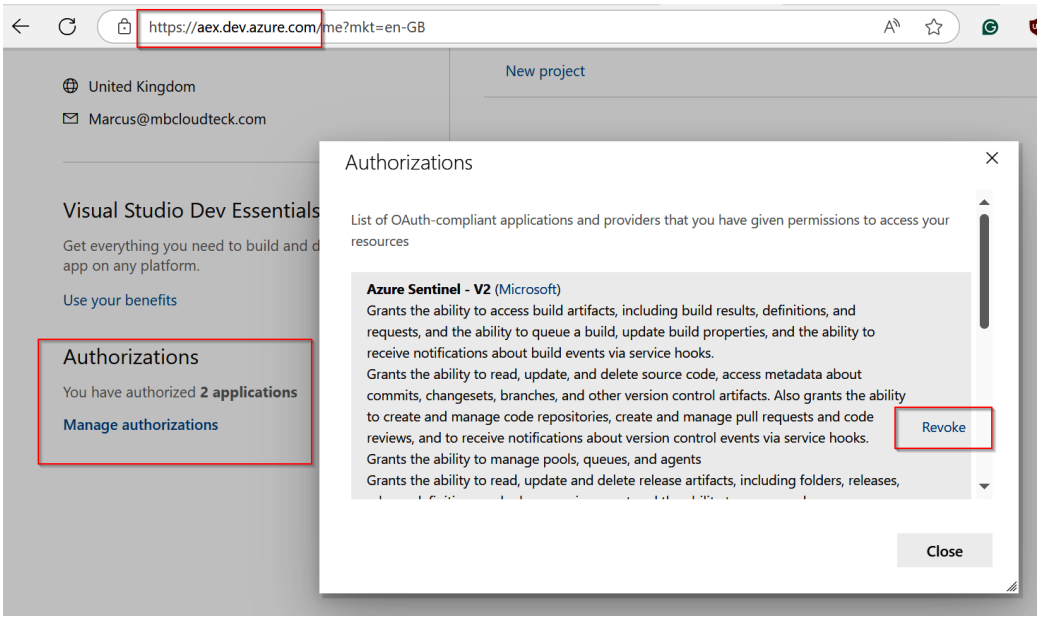


Figure 9.4: Deauthorizing or removing an existing DevOps Account authorization to connect with the right context

When you're onboarding a client or new Sentinel instance for the client, you will need that client to invite or elevate ***your*** home account that has the right access to the Azure DevOps repo, via a B2B or Lighthouse connection, temporarily to make the connection. This is yet another reason to start with a fresh subscription, making the client far more at ease with these permissive temporary rights.

After the connection is established, you can safely remove these elevated permissions from both accounts without affecting the functionality of the repository link.

Adding your repo details

Once authenticated on the setup screen, select your organization, project, repository, branch, and the types of content you plan to deploy. You can deploy everything from hunting queries to automation rules.



If you're deploying parsers or hunting content, they're bundled together via the Saved Searches API. Other content types are deployed independently, based on what you select.

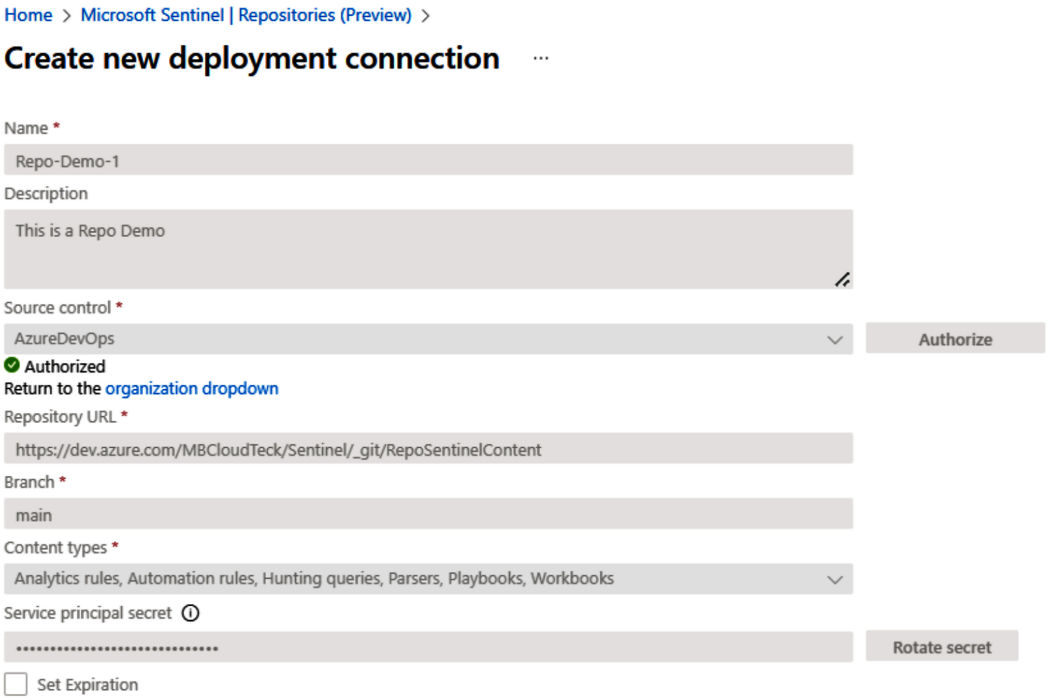


Figure 9.5: Details required to make a new repository connection, including URL, branch, content types, and service principal box

Once you enter all these details, click **Create**. This generates a new GitHub workflow or Azure DevOps pipeline that handles content deployment automatically. The amount of time this takes depends on how much content you're pushing.

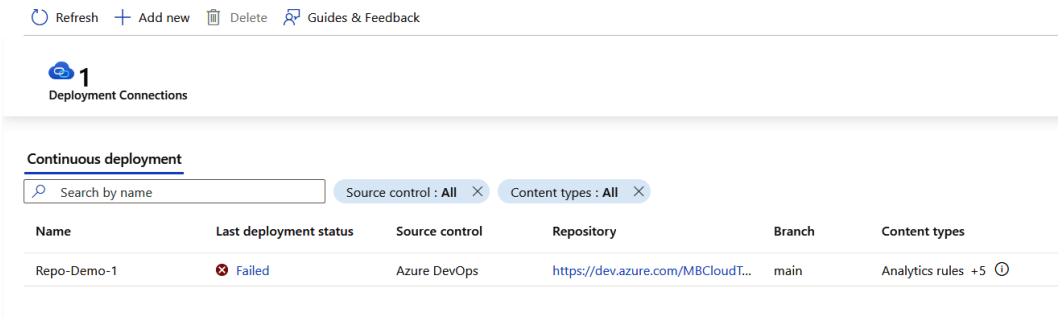


Figure 9.6: Completed connection in the Sentinel repository connection portal

Now that the Azure DevOps connection is in place, you have a trusted deployment path into Azure and Microsoft Sentinel. The next step is to make the repository ready for deployment, so your code is organized, repeatable, and safe to promote across environments.

In practical terms, repo setup is where you standardize structure, define environment boundaries (dev/test/prod), and introduce the minimum governance needed for controlled change. Once this foundation is in place, your pipelines become configuration-driven rather than bespoke, so that you can deploy content consistently, review it properly, and roll back with confidence.

Repository setup

In environments where multiple Microsoft Sentinel tenants or customer instances are managed, it's common to encounter differing deployment requirements. These differences can include tenant-specific Sentinel workspaces and resource groups, customized detection rules per tenant, unique exclusions, such as IP ranges or domains, and distinct role-based access or permissions.

While it's technically possible to use multiple Azure DevOps repositories and link them to different Sentinel instances, we'll focus on a single-repository model, which is the most manageable and scalable approach for many MSSP organizations. Once this foundation is in place, your pipelines shift from one-off, bespoke deployments to a configuration-driven model, so you can promote changes consistently, enforce reviews, and roll back cleanly when needed.

Directory versus branch-based deployment models

When using a single DevOps repository to manage multiple Sentinel deployments, you have two main structuring options: **directory-based setup** (recommended) and **branch-based setup**.

The directory-based model uses folders within the main branch to isolate configurations and content per tenant. It is easier to manage, especially when many resources are shared across tenants, and it is more suited for structured DevOps pipelines.

The branch-based model uses separate branches to isolate configuration and content per tenant (or per environment). It can work well when each deployment stream needs strong separation, independent release cadence, or different approval paths, but it becomes harder to govern at scale, especially when you need to share common content because changes must be merged and reconciled across multiple branches.

With the repository structure agreed, whether directory-based or branch-based, the next decision is how to translate that structure into real deployment control. This is where pipeline design becomes the lever: you can either tailor pipelines to each client so deployments align to their specific workspaces, exclusions, and approval paths, or you can deploy by solution unit from a shared catalogue of content.

In the next section, we'll look at two scalable patterns for doing this from a single central repo: client-specific pipelines, which prioritize clear tenant ownership and change control, and solution-driven deployment, which prioritizes reusable building blocks that can be promoted consistently across multiple clients.

Client-specific pipelines and solution-driven deployment from a central repository

In this structure, each client is treated as an independent deployment target, with its own configuration and list of supported content packages (solutions).

To support this model, we use the following:

- A central Azure DevOps repo
- A dedicated folder for each client (tenant) under `Customers/`
- Reusable solution content under `Data Connectors/`
- A per-client YAML deployment pipeline that defines what gets deployed based on "solution picker" logic

```
/
├─ .sentinel/
│  ├─ Archive/
│  └─ Customers/
│     ├─ <TenantFolderA>/
│     │  ├─ Config.yaml
│     │  ├─ sentinel-deploy-<id>.yml
│     │  └─ tracking_table.csv
│     ├─ <TenantFolderB>/
│     └─ ...
├─ Data Connectors/
│  ├─ <SolutionA>/
│  ├─ <SolutionB>/
│  └─ ...
└─ Baseline/
```

Figure 9.7: Repository structure example

As shown in *Figure 9.7*, the purpose of the three main components is as follows:

- `Customers/` contain configuration files and deployment logic scoped to each tenant
- `Data Connectors/` holds reusable solutions, such as analytics rules, playbooks, parsers, and workbooks
- `Baseline/` can be used for shared, common content across all tenants

Each tenant folder contains a unique YAML pipeline file to control the deployment process, a configuration file that lists which content folders (solutions) to deploy, and an optional tracking file for operational logging.

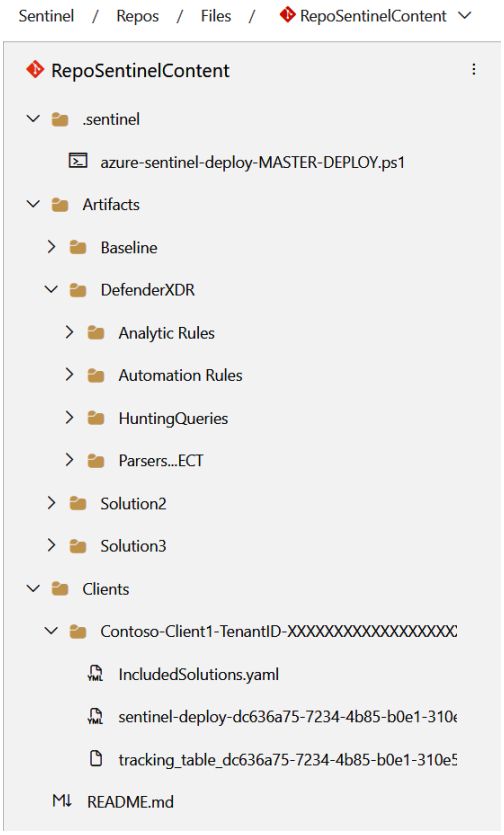


Figure 9.8: Demonstration repository structure showing the file tree with Main, Artifacts, and Clients folders

In the example shown in *Figure 9.8*, the repository is divided by purpose, mainly because **Shared content** like Defender XDR rules, playbooks, and baselines) lives in top-level solution folders and **client-specific logic** is completely contained under the `Clients/` directory, ensuring a clean separation of configuration and content.

Each tenant defines its own `IncludedSolutions.yaml` file, which acts as a solution picker list. This defines exactly which folders should be deployed for that specific tenant. This improves governance and reduces the risk of over-deploying irrelevant content. For example, one client might only require Baseline and Solution3, while another might need a broader Defender-aligned deployment.

Per-tenant CI/CD pipelines

Each client folder contains a dedicated Azure DevOps YAML file like `sentinel-deploy-xxxx.yml`, which references only the solutions listed in that tenant's picker file, points to a shared deployment script in `.sentinel/`, and supports scoped triggers so the pipeline only runs when relevant content changes. This design allows each tenant's deployment pipeline to be versioned and triggered independently, with full traceability.

Centralized reusability

All scripts and templates, including the `azure-sentinel-deploy-MASTER-DEPLOY.ps1` script, live in the `.sentinel/` directory. This promotes reuse and consistent deployment logic across clients. This also makes the onboarding of new tenants faster and less error prone.

Audit-ready

An optional tracking_table.csv file per client can be used to log what content was deployed, when the deployment occurred, and who triggered it. This can support both internal governance and client-facing reporting.

Example configuration file (Config.yaml)

This YAML acts as a solution picker, telling your deployment script or pipeline exactly which folders and files to deploy for this tenant, while skipping everything else. This is important because if a tenant has no integration with a specific data connector or solution, there is no reason to push its content.

The model mirrors the idea of not installing drivers or apps for hardware you don't own. If there's no active connection or threat vector, then there's no value and no justification for pushing that content. Each tenant receives only what's needed for their current integration and use cases.

Figure 9.9 shows an example of a solutions picker:



Figure 9.9: Client solutions selection picker list of content folders

The folders include `workspaceId`, which specifies the unique Sentinel workspace this deployment will target and `solutionsToDeploy` lists the solution folders that should be deployed for this tenant. Each entry such as, `Defender XDR`, `Solution2`, `Solution3`, and `Baseline` refers to a folder in the repo. The value `All` implies that all supported content types like analytic rules, hunting queries, workbooks within that folder should be deployed.

Example deployment pipeline file (`sentinel-deploy-<id>.yml`)

Each Microsoft Sentinel client, or tenant, in a multi-tenant deployment model should have a dedicated Azure DevOps pipeline YAML file. This allows the automation of content deployment scoped specifically to that client's workspace and required solutions. The following example demonstrates such a pipeline and forms the operational backbone for a tenant-specific, content-as-code deployment.

main / Clients / Contoso-Client1-TenantID-... / sentinel-deploy-dc636a75-7234-4b85-b0e1-310e5654ad3e.yml

sentinel-deploy-dc636a75-7234-4b85-b0e1-310e5654ad3e.yml

Contents History Compare Blame

```yaml
# Deploy Content to sentinel-demo-1 [dc636a75-7234-4b85-b0e1-310e5654ad3e]

trigger:
  branches:
    include:
    - main
  paths:
    exclude:
    - .sentinel/*
    include:
    - Clients/Contoso-Client1-TenantID-XXXXXXXXXXXXXXXXXXXX/IncludedSolutions.yaml

pool:
  vmImage: 'windows-latest'

steps:
- task: AzurePowerShell@5
  inputs:
    azureSubscription: 'Sentinel_Deploy_ServiceConnection_**********DEFAULT VALUE FROM SETUP**********'
    ScriptType: 'FilePath'
    ScriptPath: '.sentinel/azure-sentinel-deploy-MASTER POWERSHELL DEPLOYMENT SCRIPT-.ps1'
    ScriptArguments:
      -ResourceGroupName 'security-rg'
      -ContentTypes 'AnalyticsRule,AutomationRule,HuntingQuery,Parser,Playbook,Workbook'
      -WorkSpaceName 'sentinel-demo-1'
      -WorkSpaceId '401b1493-6fe2-465c-9daf-cd909ba3b91d'
      -systemAccessToken '$(System.AccessToken)'
      -branchName '$(Build.SourceBranchName)'
      -repositoryName '$(Build.Repository.Name)'
      -collectionUri '$(System.CollectionUri)'
      -projectId '$(System.TeamProject)'
      -sourceControlId 'dc636a75-7234-4b85-b0e1-310e5654ad3e'
      -rootDirectory '$(Build.SourcesDirectory)'
      -smartDeployment 'true'
    azurePowerShellVersion: 'LatestVersion'
```

Figure 9.10: YAML breakdown: sentinel-deploy-<deployment-id>.yml

This file defines the deployment pipeline for a specific Sentinel workspace using Azure DevOps. It is uniquely assigned to a client and structured to trigger only when relevant content is updated. The YAML pipeline reads the tenant's configuration and deploys only the relevant content folders as defined. Only the paths specified in the `include:` section will trigger the pipeline, and only those directories will be deployed.

Key sections of the pipeline include the following:

- **Trigger**: The pipeline is triggered when a change is committed to the main branch, but only if that change occurs in the client's `IncludedSolutions.yaml` file. This file acts as a solution picker that defines which content folders should be included in the deployment. The exclude line ensures unrelated internal logic or scripts in `.sentinel/` do not trigger deployments unnecessarily.

- **Pool**: This defines the agent pool used to run the job. In this case, the latest Windows-hosted VM is used to execute the PowerShell deployment script. Usually, this is the default when using self-hosted agents in Azure DevOps.
- **Steps**: A single job is executed using the Azure PowerShell task, which calls a master deployment script.
- **Script configuration**: The shared master deployment script is housed under the `.sentinel/` directory. This centralized script is reused across all tenant pipelines to maintain consistency and reduce duplication.

Parameters passed to the script include `ResourceGroupName` and `WorkSpaceId` point to the correct client Azure Sentinel environment. `ContentTypes` specifies the types of content to deploy. Additional arguments, such as `branchName`, `repositoryName`, and `projectId`, enable traceability and help the script determine context during execution. `smartDeployment 'true'` enables logic within the script to only deploy content that has changed or is relevant, rather than re-deploying everything.

When managing content deployments across multiple Microsoft Sentinel tenants through DevOps pipelines, the foundation typically involves scoped triggers, tenant-specific configurations, and reusable solution packages. However, depending on the maturity of your environment and the complexity of client requirements, there are several additional enhancements worth considering.

You may choose to extend your pipelines to support more granular logic, such as dynamically detecting impacted tenants based on changes in shared content. This can help optimize deployment scope and reduce unnecessary operations. It's also possible to integrate mechanisms that automate pull request feedback by flagging which tenants are affected by a proposed change, based on predefined mappings between solutions and customers.

Security and compliance are also important areas to factor in. Role-based access to deployment pipelines, secret management, and audit logging can be layered into your DevOps process to meet internal policy or regulatory obligations. Additionally, supporting parallelized or conditional deployments based on client tiers, regions, or solution types may help align with service-level agreements and operational expectations.

Finally, pipeline reusability, parameterization, and modular design will become increasingly important as your deployment footprint grows. Structuring your repository and CI/CD templates in a way that promotes consistency, scalability, and maintainability will ultimately allow your Sentinel-as-Code approach to evolve with the needs of your team and your clients.

Amending your pipeline file location

When you make the connection from Sentinel to Azure DevOps or GitHub, a connection with an action or pipeline will be created. Now, by default, this will be for that single instance, and adding content to the specified included file location (supported content added in the path included) will work. However, if you have modified the repository structure, as we have, you will also need to make changes in the pipeline section.

You will need to change the deployment YAML reference file location to select the correct file relating to that client inside the client folder.

Follow these steps to make the changes in the pipeline section:

1. In the pipeline section, select **Edit**, then **Settings**.

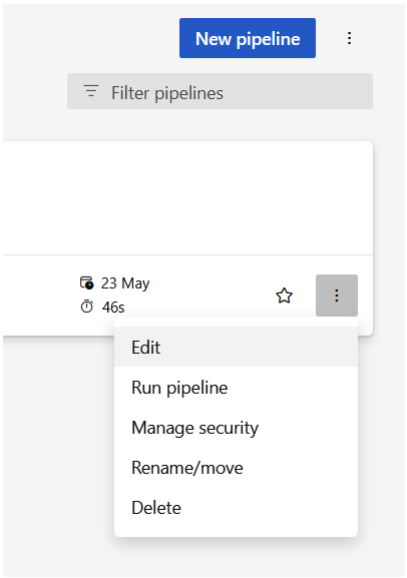


Figure 9.11: Navigating to the pipeline settings section via Edit

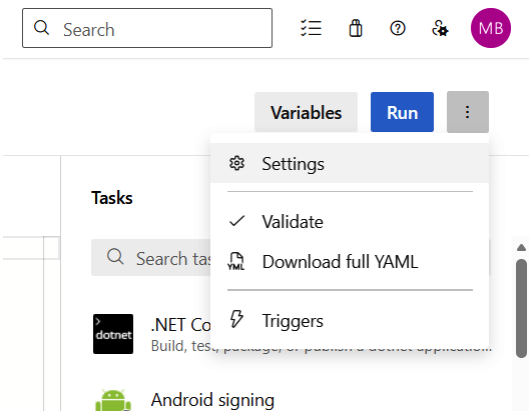


Figure 9.12: The pipeline settings cog

2. Select the correct client-specific `YAML` file.

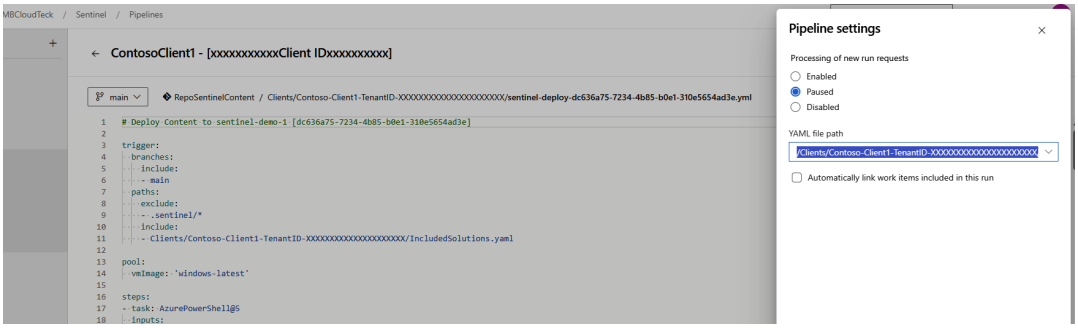


Figure 9.13: Selection of the correct pipeline client YAML file

Tracking deployment status

After setting up, you can monitor deployment results right from your repository. For GitHub, go to the **Actions** tab and open the relevant workflow file to see detailed logs. In Azure DevOps, check the **Pipelines** tab for the same information.

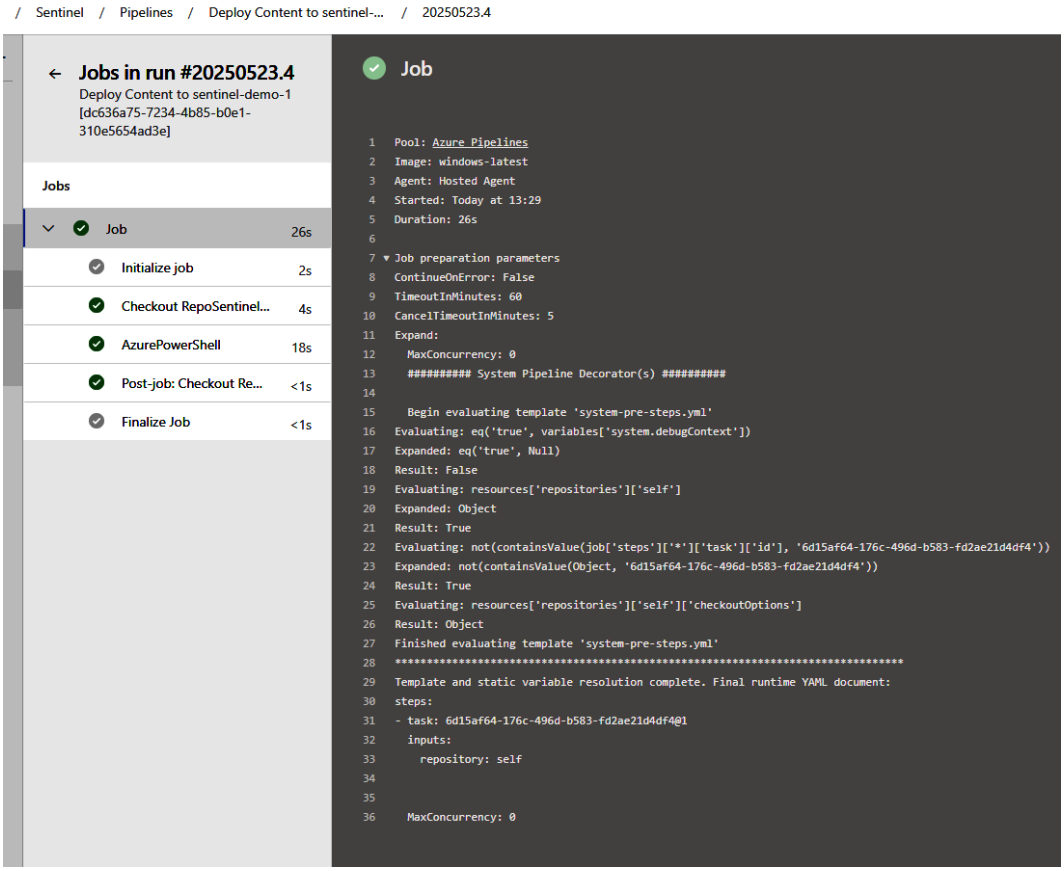


Figure 9.14: Pipeline in Azure DevOps

The key sections shown in the **Job** pipeline in Azure DevOps are as follows:

- **Initialize job**: Azure DevOps spins up the environment and prepares the pipeline for execution.
- **Checkout RepoSentinelContent**: The repository is pulled into the build agent's workspace. This includes all `.ps1` deployment scripts, YAML configuration files, and content folders like `Baseline/` and `DefenderXDR/`.
- **AzurePowerShell**: This is the heart of the pipeline. It invokes your master deployment script (`azure-sentinel-deploy-MASTER-DEPLOY.ps1`) with all required arguments like workspace ID, solution picker path, and content types. This step ran for 18 seconds, indicating a fast, efficient deployment of scoped Sentinel content.

- **Post-job: Checkout RepoSentinelContent**: This is a cleanup step, where DevOps resets the repository state or handles any final post-job logic.
- **Finalize Job**: The job wraps up, releasing the build agent and marking the pipeline run as successful.

Once a deployment is completed, your Sentinel workspace will reflect the new content. You'll also see updated connection info on the **Repositories** page, including timestamps and links to logs. Sentinel uses smart deployment logic by default; it only pushes content that's changed since the last commit. If you want to modify that behavior, you can customize the workflow to change triggers, folder scopes, or disable that delta logic entirely. To learn more, check out this link:

- `https://learn.microsoft.com/en-us/azure/sentinel/ci-cd-custom-deploy?utm_source=chatgpt.com&tabs=github`
- `https://learn.microsoft.com/en-us/azure/sentinel/ci-cd?utm_source=chatgpt.com&tabs=github`

Once a repository is connected, it becomes the source of truth for your content. If you need to make changes to something like an analytics rule, do it in the repository. Avoid editing directly in Sentinel unless you plan to export those updates back to source control, or risk having your changes overwritten on the next deployment.

Removing content works a little differently. If you delete something from the repository, it won't automatically disappear from Sentinel. You'll need to remove it manually from both locations. A useful trick is to filter content in Sentinel by the source repository name, making it easy to spot what came from where.

If you ever need to remove a repository connection, maybe you're updating to a newer deployment model or restructuring your content, the process is quick. Go to **Repositories** in Sentinel, find the connection, and click **Delete**. Confirm your choice, and the link between Sentinel and your repo is severed. However, any content already deployed remains in place unless you remove it separately. Also, if you run into issues deleting a connection, check that the associated workflow or pipeline is no longer active in your source control system.

Deploying content via Workspace Manager

The **Workspace Manager (Preview)** provides a UI-driven approach for deploying content across multiple Sentinel workspaces. It simplifies cross-workspace configuration by allowing administrators to push rules, watchlists, and automation assets from a central pane, ideal for teams managing large or complex deployments without needing to rely solely on code.

Workspace Manager is an excellent starting point for teams looking to deploy Microsoft Sentinel content quickly across multiple workspaces without needing to build automation pipelines from day one. It provides a simple, scalable interface for managing assets centrally. As your security operations mature, this approach can be seamlessly evolved into a repository-based, version-controlled deployment model, offering greater flexibility, auditability, and CI/CD integration over time.

Before diving in, make sure you've got at least two Sentinel workspaces: one to act as the central hub, and one or more to serve as managed members. You'll need Microsoft Sentinel Contributor access on both the central workspace and any member workspaces you want to control. If you're managing workspaces across different Azure AD tenants, you'll also need to configure Azure Lighthouse. This is essential for gaining delegated access across tenant boundaries.

There's no one-size-fits-all layout. Depending on how your organization operates, you might lean toward one of the following:

- **Direct-Link**: A simple structure where a single central workspace manages all member workspaces directly.
- **Co-Management**: Useful when both internal SOC teams and MSSPs need to manage the same set of workspaces. Multiple central workspaces share control over the same members.
- **N-Tier**: Designed for more complex setups, think of a parent organization managing subsidiaries, each with its own internal workspaces. One central workspace oversees others, which in turn manage their own environments.

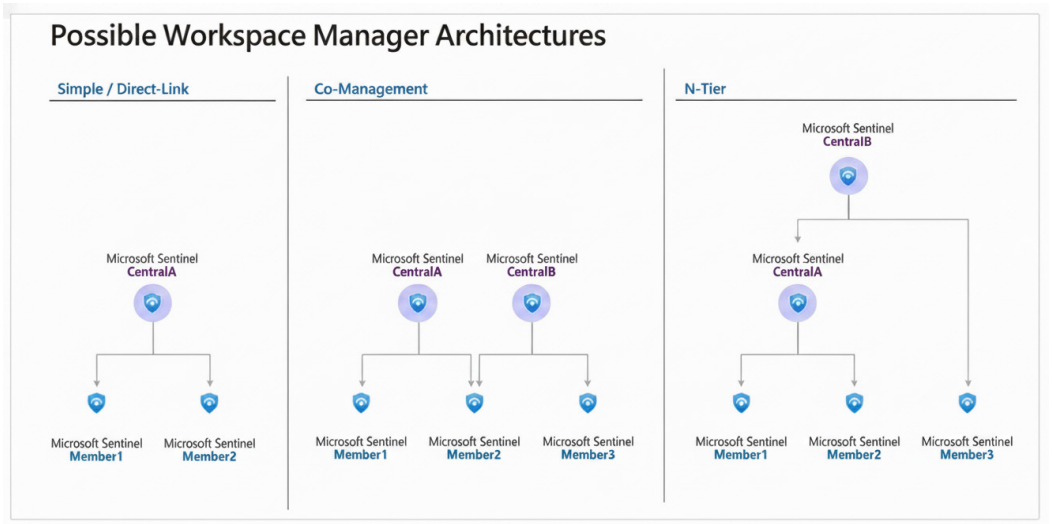


Figure 9.15: Possible Workspace Manager architectures

Once you've selected your central workspace, enabling Workspace Manager is straightforward.

Go to the **Settings** blade in that workspace and toggle the option to **Make this workspace a parent**.

Once enabled, you'll see a new **Workspace Manager (Preview)** option under the **Configuration** menu.

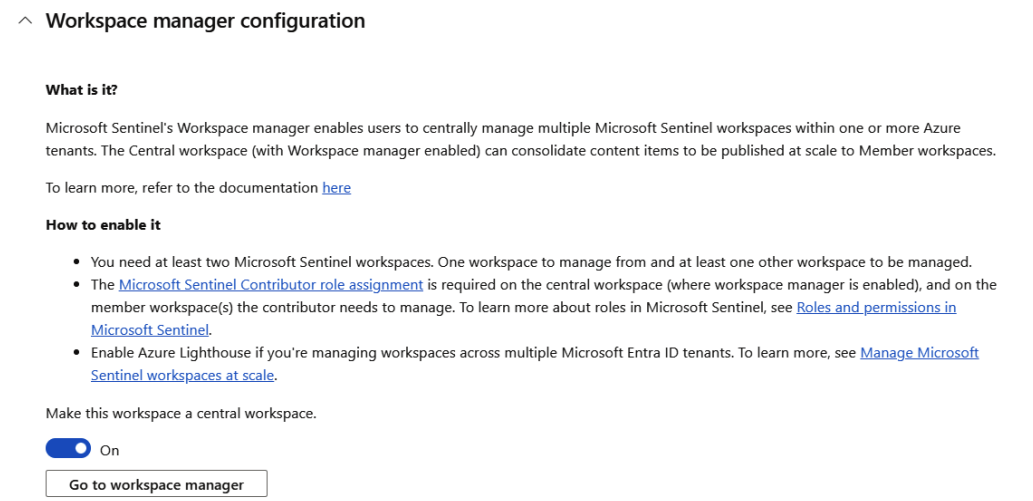


Figure 9.16: Enabling Workspace Manager within Sentinel settings

From there, you can start onboarding other Sentinel workspaces. Just select **Add workspaces** and pick the member workspaces you'd like to manage. If Azure Lighthouse is in place, you can even onboard workspaces from other tenants.

Organizing with groups

Groups in Workspace Manager help you organize your workspaces logically by region, business unit, vertical, or any structure that makes sense to your team. This grouping becomes especially useful when you're deploying targeted content sets across different segments of the organization.

Follow these steps to set up a group:

- For a single workspace, select **Add | Group**
- For multiple, select the workspaces and then **Add | Group from selected**

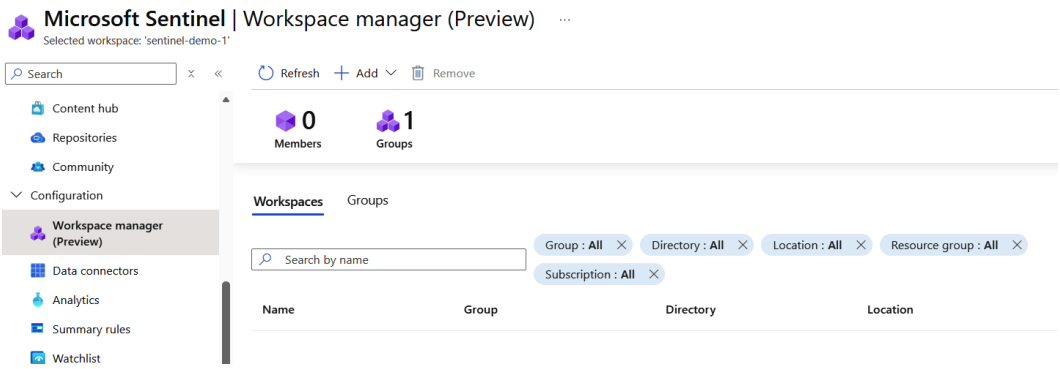


Figure 9.17: Group definition creation in Workspace Manager – Add button

When creating the group, add name and description, then select which workspaces belong to it.

On the content side, you've got two options:

- **All Content**: Pulls in everything currently active in the central workspace (this is a snapshot, not linked templates).
- **Selected Content**: Lets you pick specific items perfect for tailored deployments.

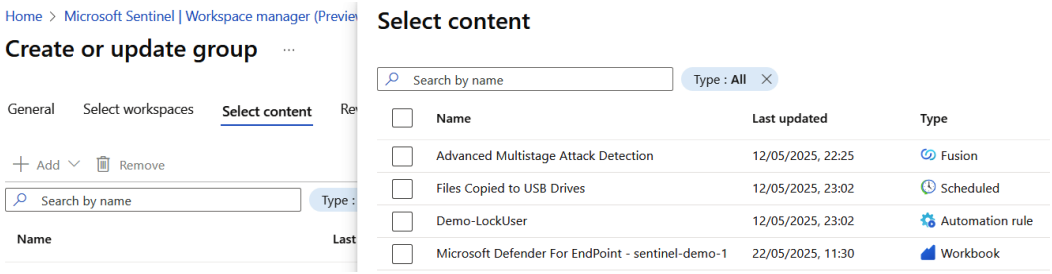


Figure 9.18: Selecting supported content to push in Workspace Manager

Once your group is created, it will appear in the **Groups** tab, ready for publishing. At this point, you've grouped your workspaces and selected your content, but it hasn't gone live yet.

Publishing content

Creating a group does not automatically deploy content. You must publish the group to push the selected content to the assigned member workspaces. Follow the wizard on screen and publish the group as shown in *Figure 9.19*.

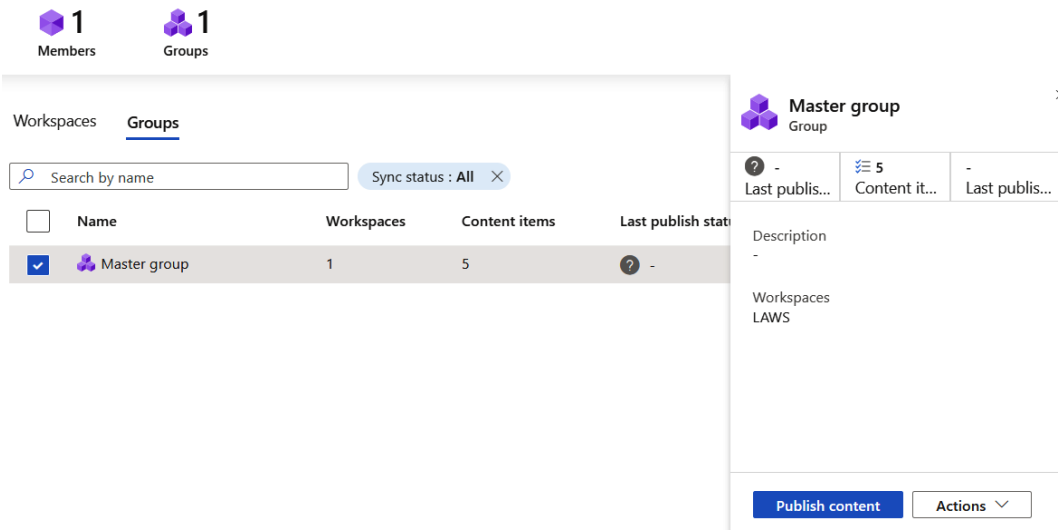


Figure 9.19: Publish content via the select groups

Publishing applies the snapshot of content as it exists at that moment in the central workspace. Any changes made after publishing require re-publication.

Workspace Manager (Preview) comes with a few important caveats:

- Playbooks linked to rules are not supported: If a playbook is tied to an analytics or automation rule, it must be manually managed outside Workspace Manager.
- Workbooks using custom storage accounts are not supported: Only workbooks stored within the central workspace are deployable. Bring Your Own Storage (BYOS) models require manual management.
- Content must originate from the central workspace: Any rules or workbooks created directly within a member workspace are not visible or manageable via Workspace Manager. You must recreate them centrally.
- No content deletion support: Currently, you cannot delete content from member workspaces through Workspace Manager. Cleanup must be done manually or through the standard Sentinel portal.
- 2000-action limit: There is a practical limit on how much content can be deployed in a single action, often sufficient for 5–8 standard workspace instances. Larger deployments may need to be broken into multiple groups or staged. For instance, in the field, we see standard Sentinel workspaces deploying between 2 and 400 rules per instance. This could hit the limit with fewer than 7 connected workspaces.

Now, let's move on to the next section, where we will show you how to fast-track your IAC Sentinel deployment.

Fast-track your IAC Sentinel deployment

If you're working in a fast-paced environment where deploying Microsoft Sentinel is a frequent task, such as in MSSP operations or within a mature Azure DevOps ecosystem, speed and consistency are critical. In these scenarios, every click matters, and automation is already your best option. This section is particularly relevant for teams that regularly provision Sentinel workspaces or operate within DevOps-centric environments where Infrastructure as Code (IaC) is the standard.

Start with infrastructure, not content

Before diving into Sentinel content deployment (which we've covered in previous sections), it's important to consider how you'll deploy the infrastructure layer itself. This includes the Sentinel workspace, log analytics, resource group, and diagnostic settings. While IaC tools such as **Terraform** and **Bicep** offer repeatability and version control, deploying Sentinel via code requires a few nuanced considerations as some components such as connectors are still manual.

Terraform offers robust modularity and support across teams but requires managing a state file, typically stored in an Azure storage account or backend such as Azure Blob Storage. Ensuring this is properly configured is key to avoiding drift or state conflicts. Bicep is more native to Azure and easier to integrate with Azure DevOps pipelines without requiring state management. It's a good choice for teams that favor Azure-native tooling and want fast iteration with ARM template parity. If your organization is already using IaC to manage Azure resources, aligning Sentinel deployment to that model can streamline operations and improve lifecycle governance.

Sentinel deployment via code

Microsoft Sentinel deployment via code is the practice of managing detections, automation, and supporting configuration as version-controlled artifacts, then releasing them through CI/CD rather than clicking changes in the portal. It gives you repeatability, peer review, and a clear audit trail, making it far easier to scale deployments across environments or multiple tenants while keeping change controlled and consistent.

Keep in mind that tearing down and redeploying the environment may not fully preserve third-party data collection configurations, many of which still require manual steps. So, while IAC is great for initial deployments, the benefits can be limited beyond that unless you're automating the full lifecycle.

Given these considerations, for most organizations operating a single Microsoft Sentinel instance, we generally do not recommend investing significant time in an IaC deployment. The return on effort is often limited unless you're managing multiple environments, require strict version control, or already have a mature DevOps pipeline in place.

Manual (All-in-One v2)

As we have explained a manual deployment in the previous sections, let's speak about the fastest manual way to deploy without any pipeline or code. If you prefer to learn by doing or simply want to get hands-on and follow along in real time, we highly recommend spinning up a development environment using the All-in-One v2 deployment tool.

For IT administrators tasked with evaluating Microsoft Sentinel, time and clarity are essential. The All-in-One tool removes the complexity of setup and delivers a fully operational Sentinel environment within minutes. This rapid deployment allows you to get straight to the part that matters, understanding how Sentinel can add value to your security operations.

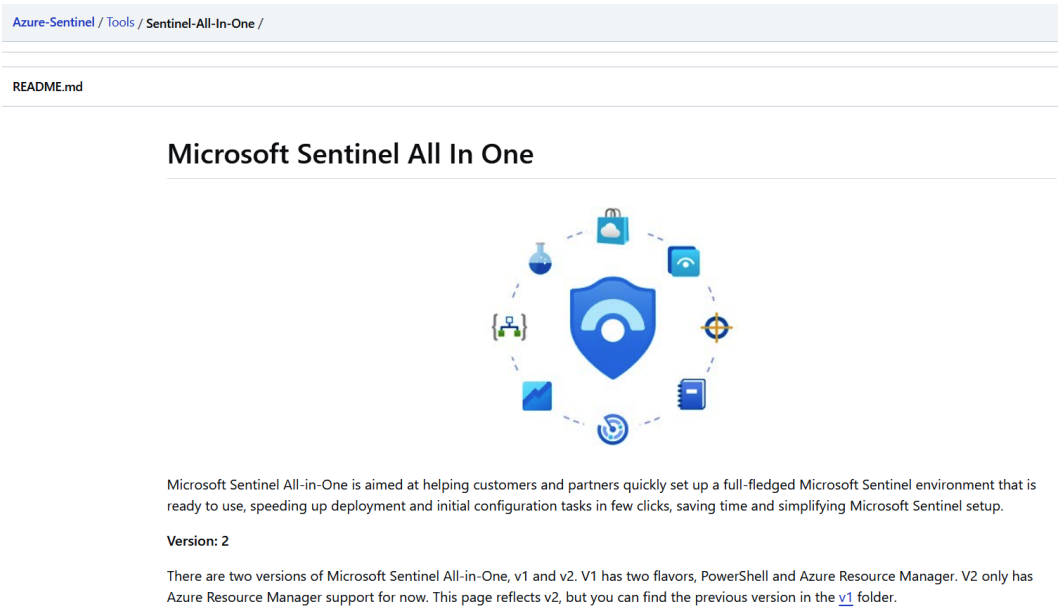
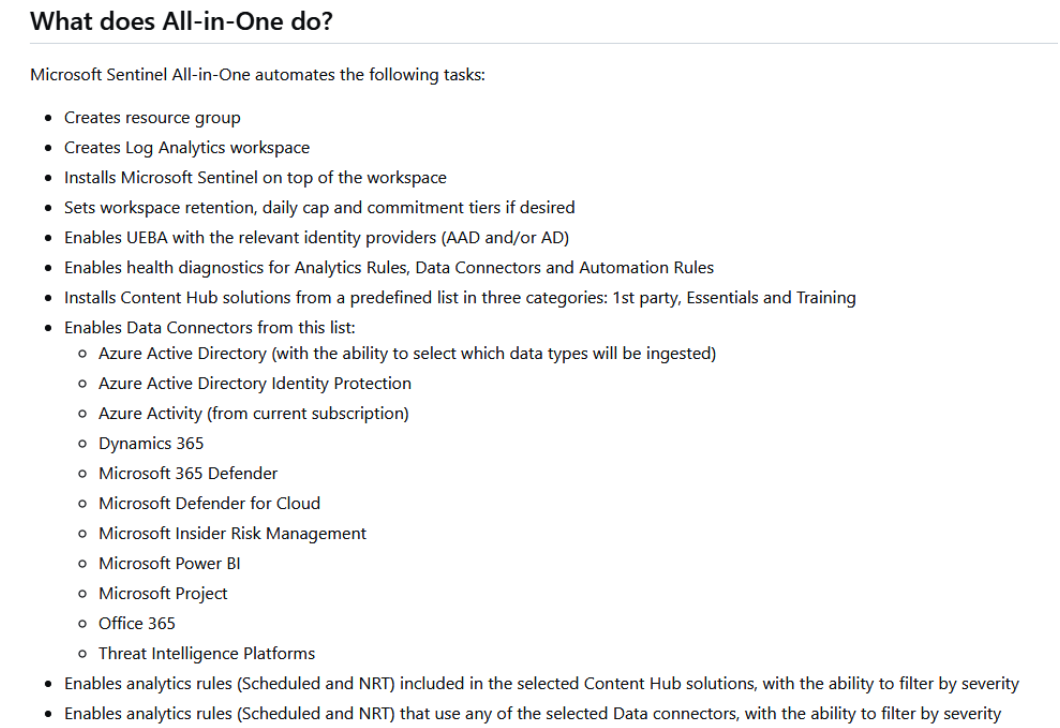


Figure 9.20: The Sentinel All-In-One v2 GitHub page

Rather than spending hours configuring resource groups, workspaces, connectors, and analytics rules, this tool will handle the foundational work automatically. As an IT admin, this means you can start exploring real-world use cases, data flows, and detections almost immediately.

Whether you're assessing Sentinel for broader adoption, preparing for a workshop, or need a clean environment to demonstrate to internal stakeholders, this tool offers a seamless starting point. It's an ideal way to see Sentinel in action, without the delays or distractions.



What does All-in-One do?

Microsoft Sentinel All-in-One automates the following tasks:

- Creates resource group
- Creates Log Analytics workspace
- Installs Microsoft Sentinel on top of the workspace
- Sets workspace retention, daily cap and commitment tiers if desired
- Enables UEBA with the relevant identity providers (AAD and/or AD)
- Enables health diagnostics for Analytics Rules, Data Connectors and Automation Rules
- Installs Content Hub solutions from a predefined list in three categories: 1st party, Essentials and Training
- Enables Data Connectors from this list:
  - Azure Active Directory (with the ability to select which data types will be ingested)
  - Azure Active Directory Identity Protection
  - Azure Activity (from current subscription)
  - Dynamics 365
  - Microsoft 365 Defender
  - Microsoft Defender for Cloud
  - Microsoft Insider Risk Management
  - Microsoft Power BI
  - Microsoft Project
  - Office 365
  - Threat Intelligence Platforms
- Enables analytics rules (Scheduled and NRT) included in the selected Content Hub solutions, with the ability to filter by severity
- Enables analytics rules (Scheduled and NRT) that use any of the selected Data connectors, with the ability to filter by severity

Figure 9.21: Details of what the Sentinel All-In-One deployment does

Here is the link where you can find the All-in-One tool:

`https://github.com/Azure/Azure-Sentinel/tree/master/Tools/Sentinel-All-In-One`



We recommend you run this with Subscription Owner and Global Admin or Security Admin for the best deployment, and one click connector standup for things Entra and Defender.

Customization and production readiness

While the All-in-One v2 deployment tool offers an excellent way to quickly spin up a working Microsoft Sentinel environment, which is perfect for testing, hands-on learning, and quick proof of concepts, it's important to understand its limitations.

The configurations, content, and analytics rules deployed by default are not tailored to your organization's unique environment, risk posture, or operational needs. These out-of-the-box artifacts should be treated as templates only, not ready-made production solutions. They provide a solid starting point for exploration and inspiration, but do not reflect the depth of customization required for an operational SOC.

In live environments, key considerations such as data governance, role-based access, retention policies, connector scopes, threat modeling, and use-case-driven analytics require deliberate architectural planning. We'll explore these elements in detail in an upcoming section focused on production-grade design.

Cost is another critical factor. The convenience of rapid deployment can obscure potential ingestion costs if data sources are connected without scoping or filtering. Due to the simplicity of deployment, we will not give the exact steps here. Head over to the previous link and deploy the All-in-One tool in a test environment today!

Sentinel deployment via Bicep

Bicep is Microsoft's native Iac language for Azure, designed to deploy and manage resources consistently using declarative templates. In a Microsoft Sentinel context, Bicep is especially useful for standardizing the platform layer around Sentinel, resource groups, Log Analytics workspaces, Sentinel enablement, data connectors (where supported), automation prerequisites (Logic Apps, identities, permissions), and supporting components such as Key Vault and Azure Storage.

Its value is control and repeatability. You can stand up or replicate Sentinel environments reliably across subscriptions, tenants, and regions, keep changes peer-reviewed in source control, and align deployments to governance requirements through predictable, testable releases. Using Azure DevOps pipelines, the project combines Bicep templates for infrastructure with a powerful PowerShell script for deploying Sentinel solutions, analytics rules, and workbooks.

Setting up the project in Azure DevOps

Let's walk through the detailed process of setting up this project in Azure DevOps.

Step 1: Creating a new Azure DevOps project

Follow these steps to create a new Azure DevOps project:

1. Log in to your Azure DevOps organization and click on **New project**.
2. Enter a project name (e.g., **Sentinel-Deployment**).
3. Choose visibility (private) and click **Create**.

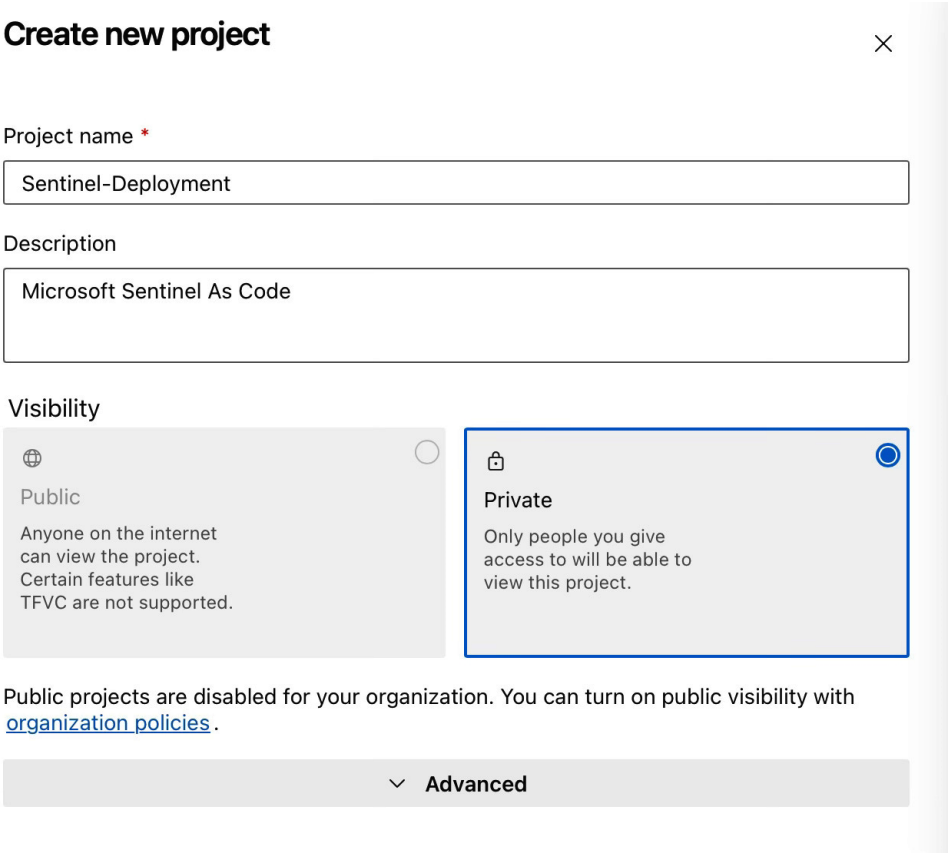


Figure 9.22: New project creation in Azure DevOps

Step 2: Importing the repository

To import the repository, follow these steps:

1. Navigate to **Repos** in your new project.
2. Click **Import** under the repository dropdown.
3. Enter the source URL or upload a `ZIP` file of the repository: `https://github.com/PacktPublishing/Microsoft-Sentinel-Blueprint.git`

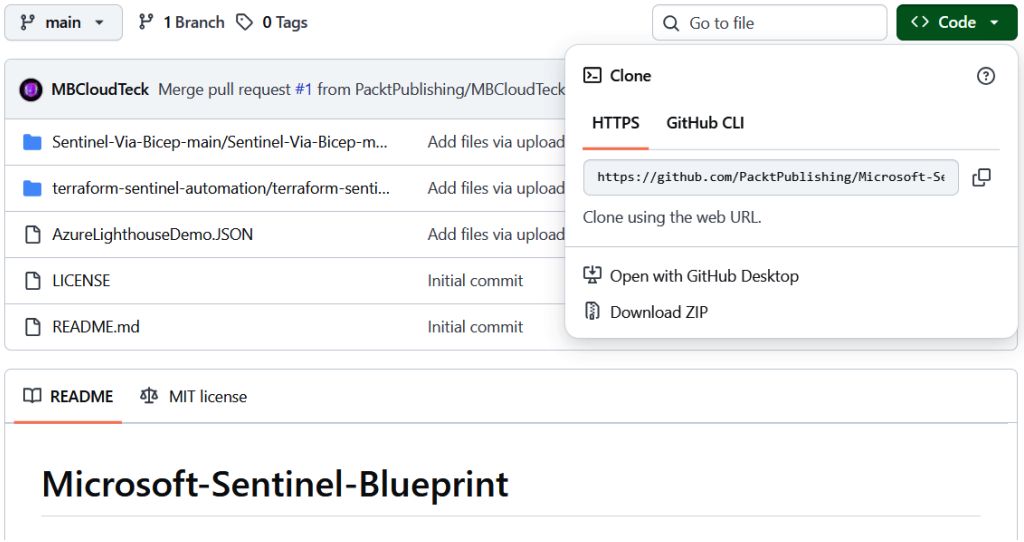


Figure 9.23: Source URL – finding the URL in GitHub

4. Click **Import.**

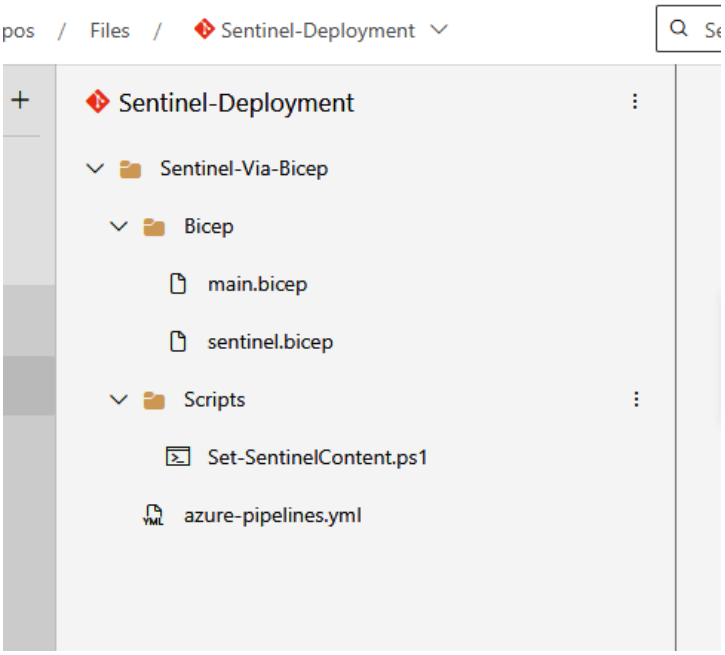


Figure 9.24: New Azure DevOps repo structure

Step 3: Creating the resource group in Azure

Before setting up the service connection, you need to create a resource group that will contain our Sentinel resources:

1. Log in to the Azure portal (`https://portal.azure.com`) and navigate to **Resource groups** in the left navigation.
2. Click **+ Create** and select your subscription.
3. Enter **SentinelRG** (or your preferred name) for the resource group.
4. Select your region (e.g., **UK South**).
5. Add your Azure resource tags (if applicable) and click **Review | Create.**

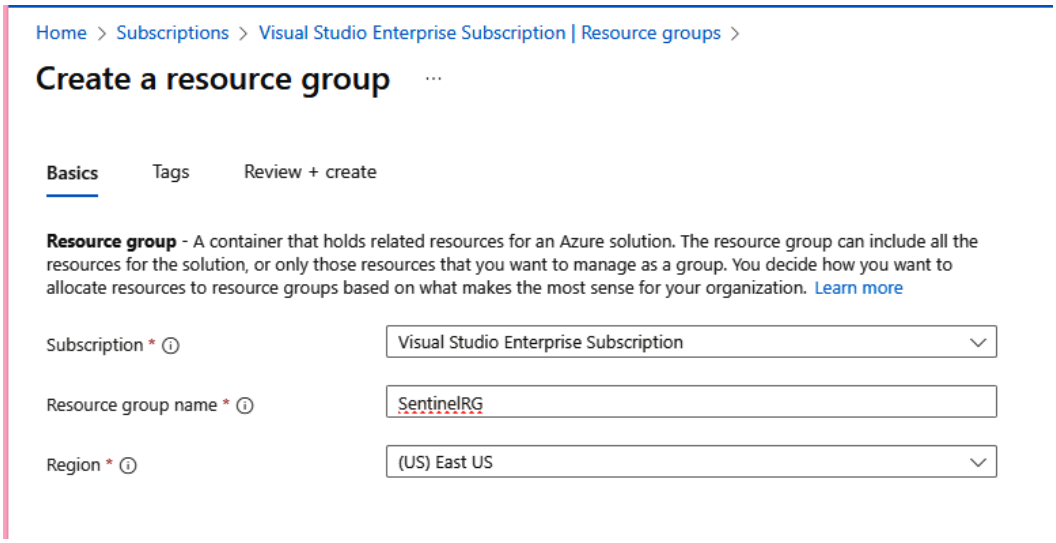


Figure 9.25: Create Azure resource group

Step 4: Configure Azure service connection

A service connection is required to deploy resources to Azure. Let's set this up:

1. Navigate to **Project settings** (bottom-left corner).

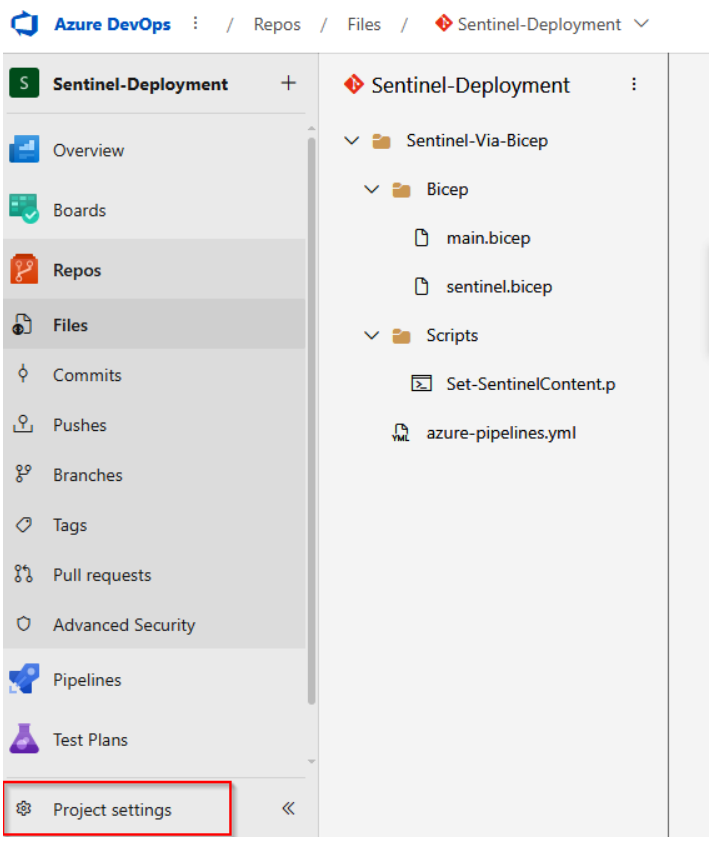


Figure 9.26: Azure DevOps Project settings

2. Select **Service connections** under **Pipelines**.
3. Click **New service connection**.

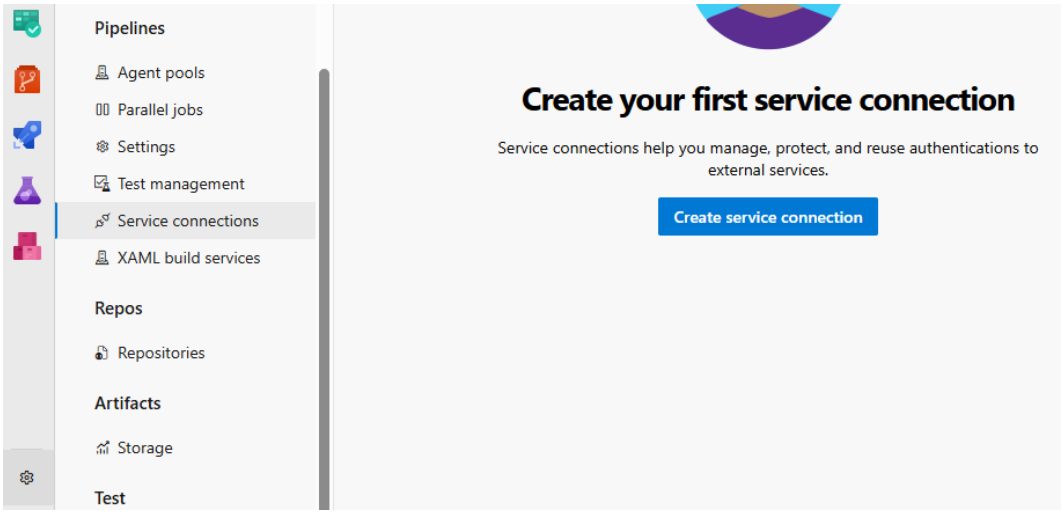


Figure 9.27: Create a new service connection in Azure DevOps

4. Select **Azure Resource Manager**.

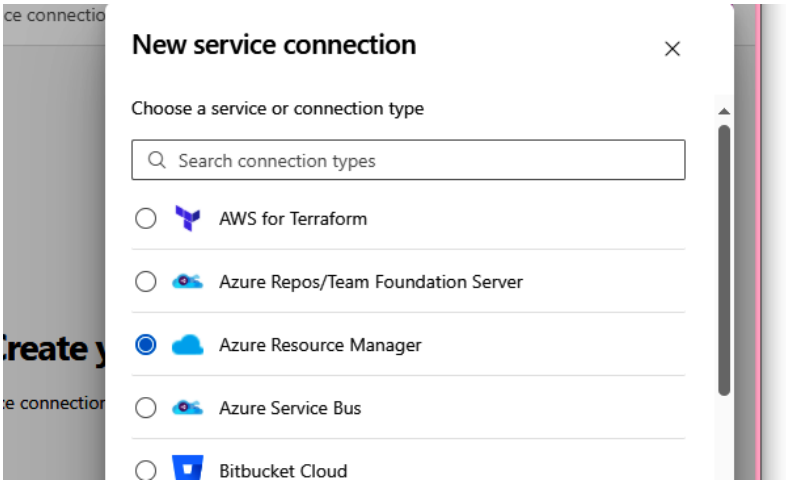


Figure 9.28: Choose your new service connection

5. Choose **Service principal** (automatic) for the authentication method.
6. Select **Scope level** as **Subscription**.
7. Select your subscription.
8. Select the **SentinelRG** resource group you created previously.
9. For the service connection name, enter **DeploymentConnection**.

10. Ensure **Grant access permission to all pipelines is** not selected (we'll revisit this later).
11. Click **Save**.

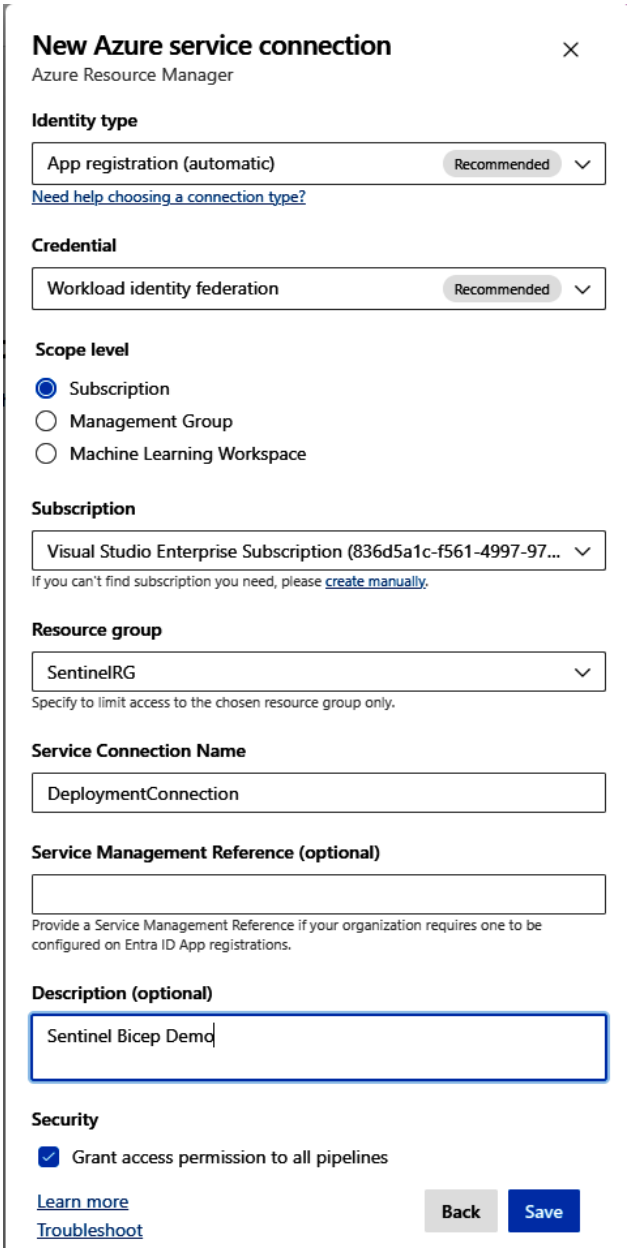


Figure 9.29: Creating a new pipeline in DevOps



If your organization requires more restricted permissions, you can create a custom service principal in Entra and assign the necessary permissions (Contributor to the target subscription or resource groups), then input this into the pipeline details.

Step 5: Configure an Entra ID Security Admin service connection

While the resource-group-scoped service connection is sufficient for basic Sentinel deployment, certain advanced features, such as UEBA and the upcoming data connectors feature (yet to be implemented), require directory-level permissions. Here's how to set up the service connection with the necessary permissions:

After creating the service connection, you need to assign the Security Administrator role in Entra ID. Follow the appropriate method based on your organization's configuration:

Standard role assignment without PIM (insecure and not recommended)

If your organization does not use Privileged Identity Management, do the following:

1. Navigate to the Microsoft Entra admin center: `https://entra.microsoft.com`.
2. Go to **Identity | Applications | Enterprise applications**.
3. Search for the service principal that was created.

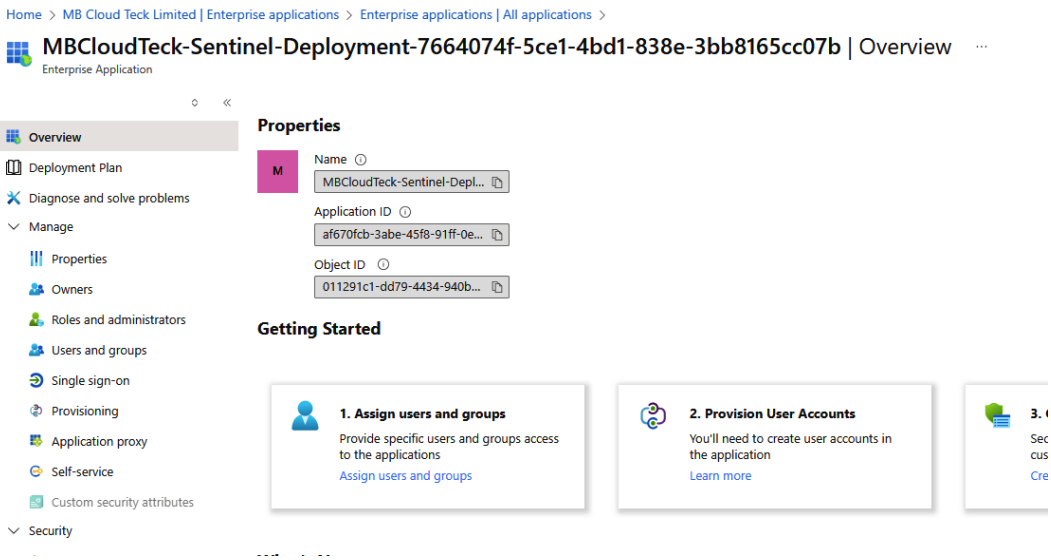


Figure 9.30: Entra ID app registration for service principal

4. Navigate to the service principal properties.
5. Select **Roles and administrators**.
6. Click **+ Add assignment**.
7. Search for **Security Administrator**, select the role and click **Add**.

Role assignment using PIM

If your organization uses PIM for just-in-time role activation, do the following:

1. Navigate to the Microsoft Entra admin center (`https://entra.microsoft.com`).
2. Go to **Identity Governance | Privileged Identity Management**.
3. Select **Microsoft Entra roles**.
4. Click on **Security Administrator** from the roles list.

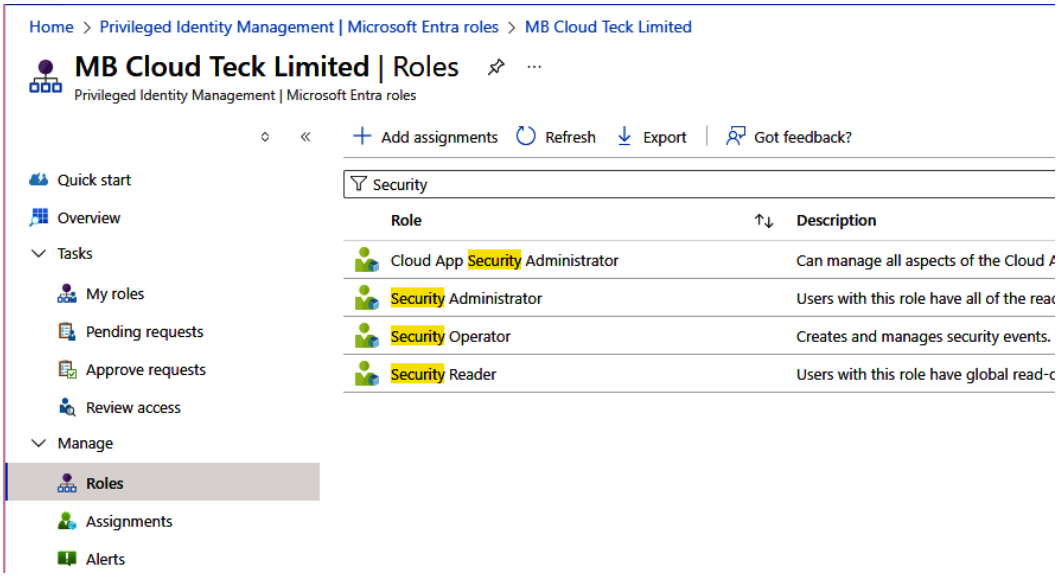


Figure 9.31: Roles required to add to the service principal

5. Select **Assignments | + Add assignments**.
6. Search for your service principal (the name will look like this: `azure devops Sentinel Deployment 00000000 0000 0000 0000 000000000000`).

7. Select the service principal.

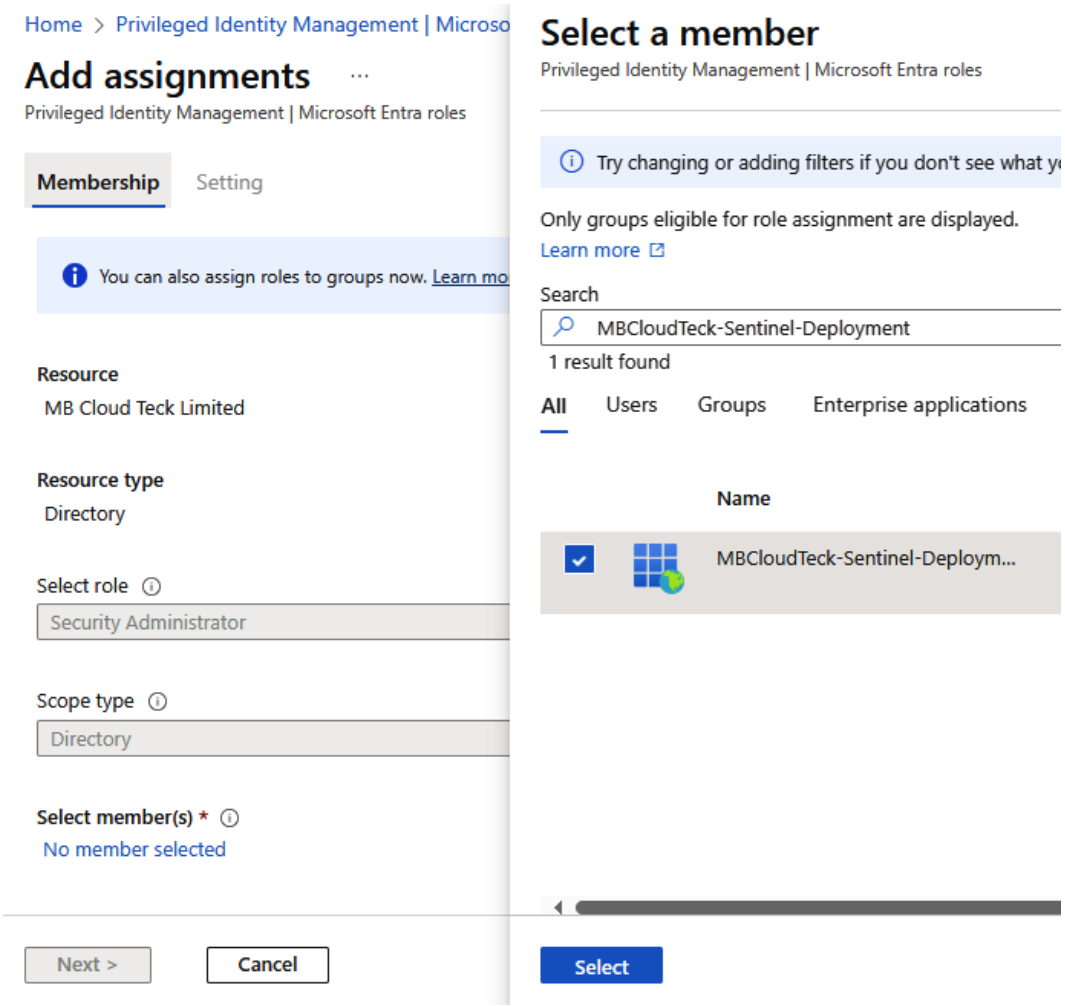


Figure 9.32: Adding role assignment to service principal

8. Under **Assignment type**, select Active (not Eligible).
9. Check **Permanently assigned** and provide a justification, such as **Deploying Via Bicep**.

10. Click **Assign**.

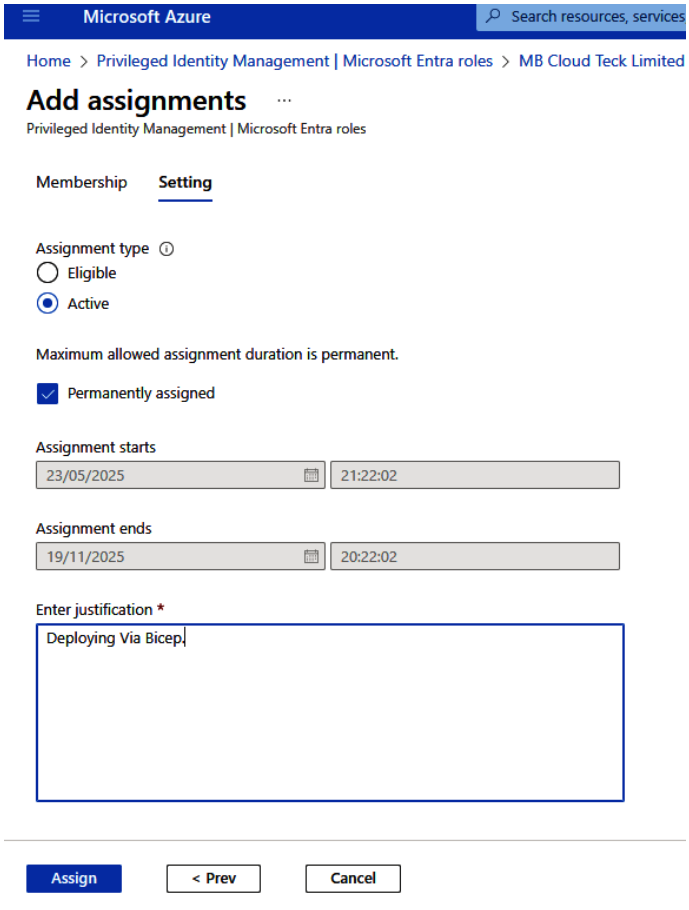


Figure 9.33: Add assignment in PIM



Unlike user accounts, service principals cannot interactively request role activation, so they must have permanently active assignments in PIM environments.

Step 6: Creating and assigning a custom RBAC role for deployments

Even with the resource group scoped service connection, we need to ensure our pipeline service principal has the necessary permissions to deploy resources using ARM/Bicep templates. Let's create a custom RBAC role with minimal permissions:

1. Save the following JSON to a file. This is also available in the GitHub repo: `https://github.com/PacktPublishing/Microsoft-Sentinel-Blueprint/blob/main/Sentinel-Via-Bicep-main/Sentinel-Via-Bicep-main/Sentinel-Via-Bicep/AzureDevOpsRole.json`

```
{
  "properties": {
    "roleName": "Azure DevOps Least Privileged Deployment",
    "description": "Minimal permissions to deploy Azure resources
using ARM templates",
    "assignableScopes": [
      "/subscriptions/YOUR SUBSCRIPTION ID HERE"
    ],
    "permissions": [
      {
        "actions": [
          "Microsoft.Resources/deployments/read",
          "Microsoft.Resources/deployments/write",
          "Microsoft.Resources/deployments/validate/action",
          "Microsoft.Resources/deployments/operationStatuses/read"
        ],
        "notActions": [],
        "dataActions": [],
        "notDataActions": []
      }
    ]
  }
}
```

2. Replace **YOUR SUBSCRIPTION ID HERE** with your actual subscription ID.

3. Create the custom role using Azure CLI: `az role definition create --role-definition AzureDevOpsRole.json`

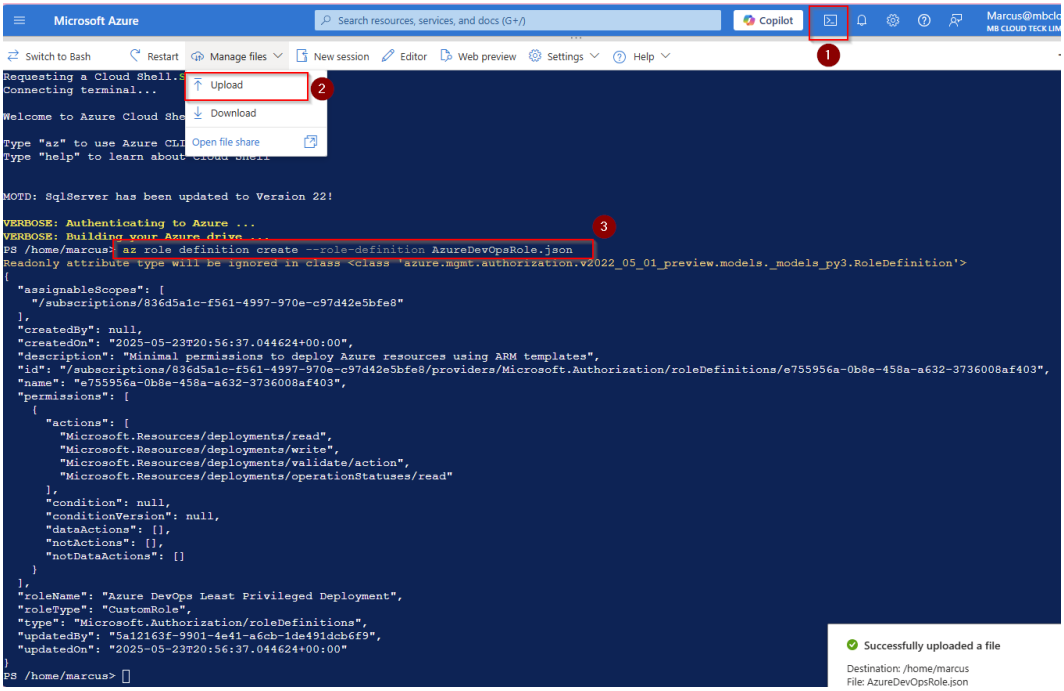


Figure 9.34: Showing the script to get from GitHub AssignLeastPrivilegedRole.ps1

Even though our service principal has Contributor access to the target resource group, it needs subscription-level permissions to initiate deployments. This custom role provides those permissions without granting excessive access to other resources in the subscription.



Azure Policy note: If you're using Azure Policy with strict enforcement, you might need to add additional permissions to this role based on policy requirements. Monitor your pipeline logs for permission-related errors and adjust accordingly.

Step 7: Creating pipeline variables

To create pipeline variables, navigate to Pipelines and follow these steps:

1. Create a new pipeline based on the Azure `pipelines.yml` file.

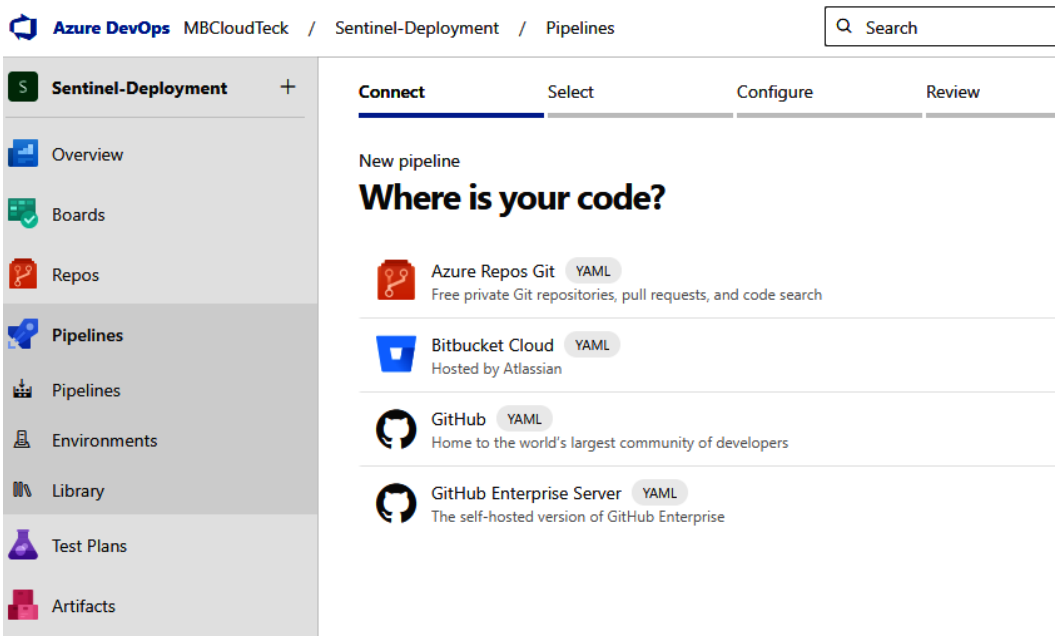


Figure 9.35: Adding a new pipeline in Azure DevOps

2. Now, select the pipeline type:

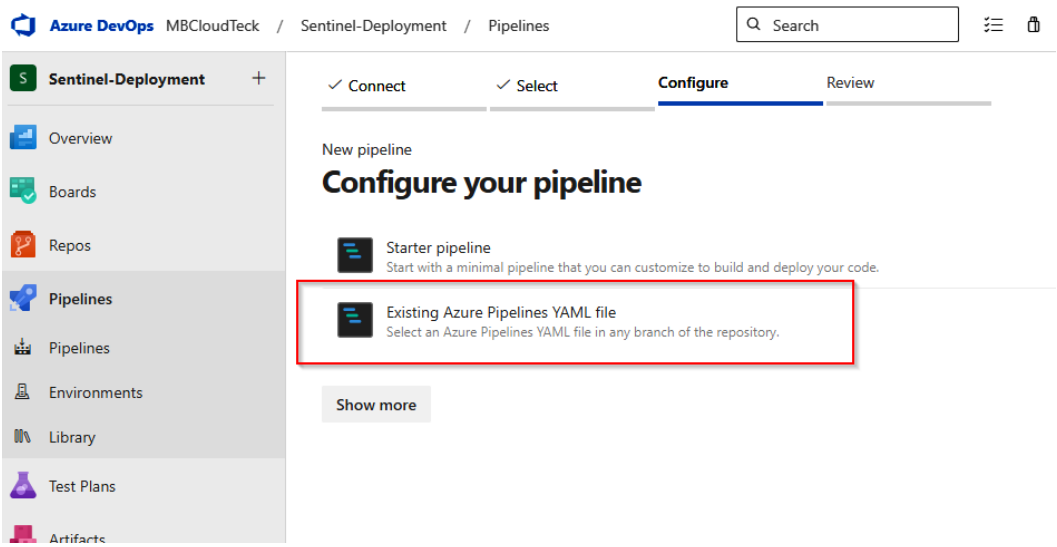


Figure 9.36: Select your YAML file in Azure DevOps

3. Now, select the correct YAML file.

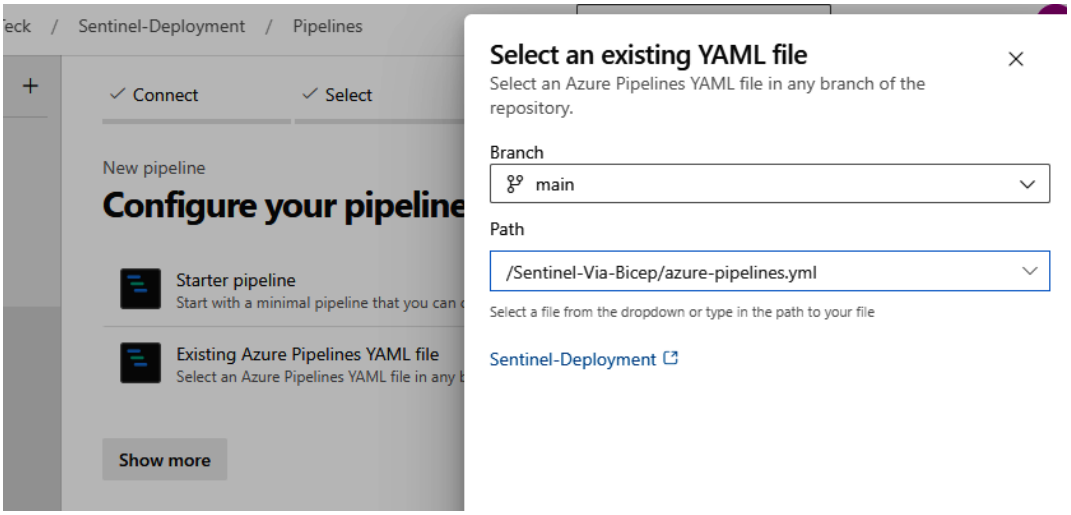


Figure 9.37: Select Branch and Path

4. Save the changes. Click **Variables** in the pipeline editor.

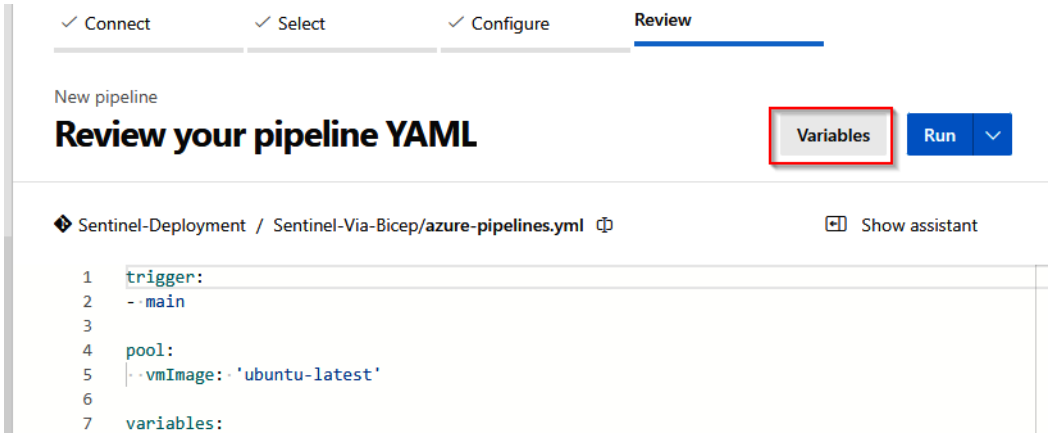


Figure 9.38: Review your pipeline variables

5. Add the following variables:

    - **RESOURCEGROUP: SentinelRG**
    - **WORKSPACENAME: SentinelLAW**
    - **REGION: uksouth**
    - **DAILYQUOTA: 0**
    - **SENTINELSOLUTIONS: “Azure Activity”, “Microsoft 365”, “Threat Intelligence”**
    - **ARSEVERITIES: “High”, “Medium”, “Low”**

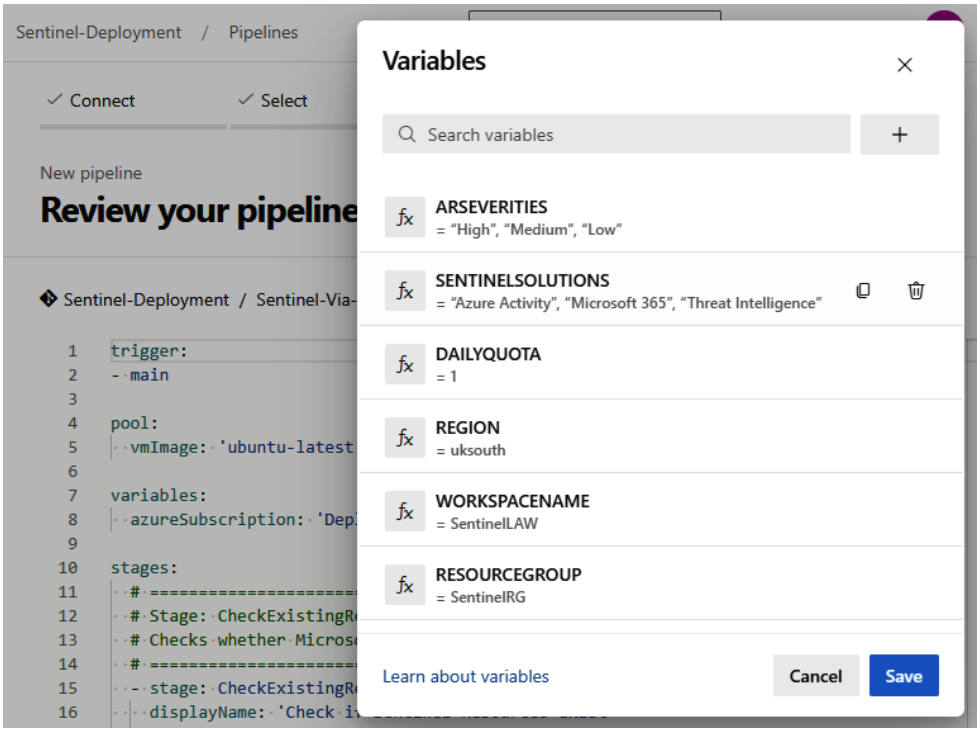


Figure 9.39: Pipeline variables

Step 8: Saving and running the pipeline

It's now time to save and run the pipeline:

1. Click the **Save** button.
2. You will receive a warning on the pipeline that it needs authentication against the service principal we created earlier. Click to view and permit the authentication.
3. Review the progress of each stage.

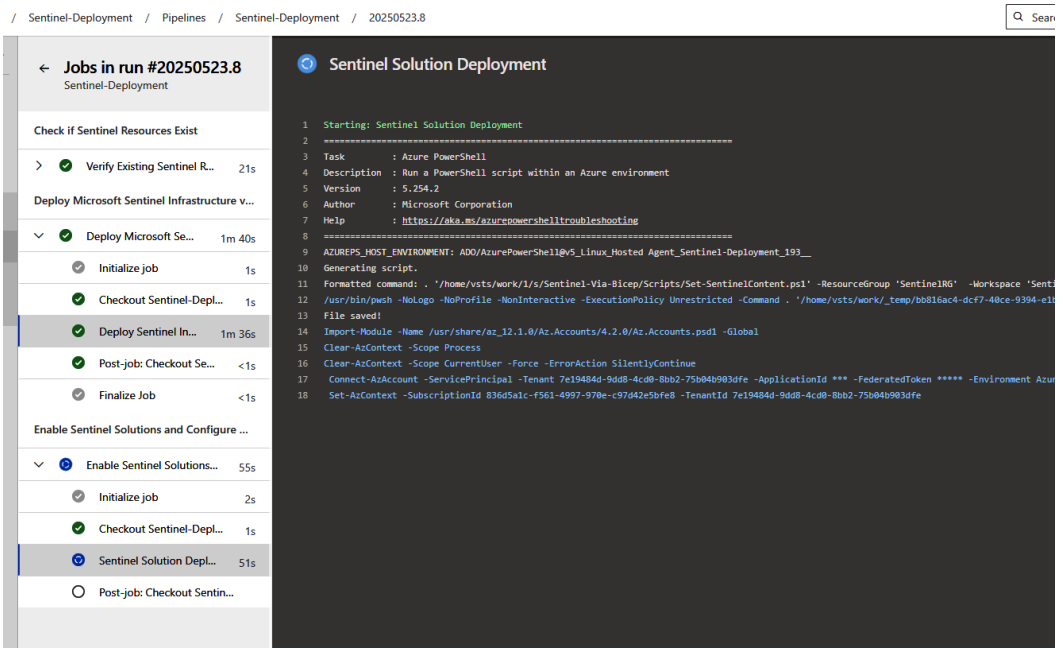


Figure 9.40: Pipeline deployment in progress

4. Verify successful deployment in the Azure portal.

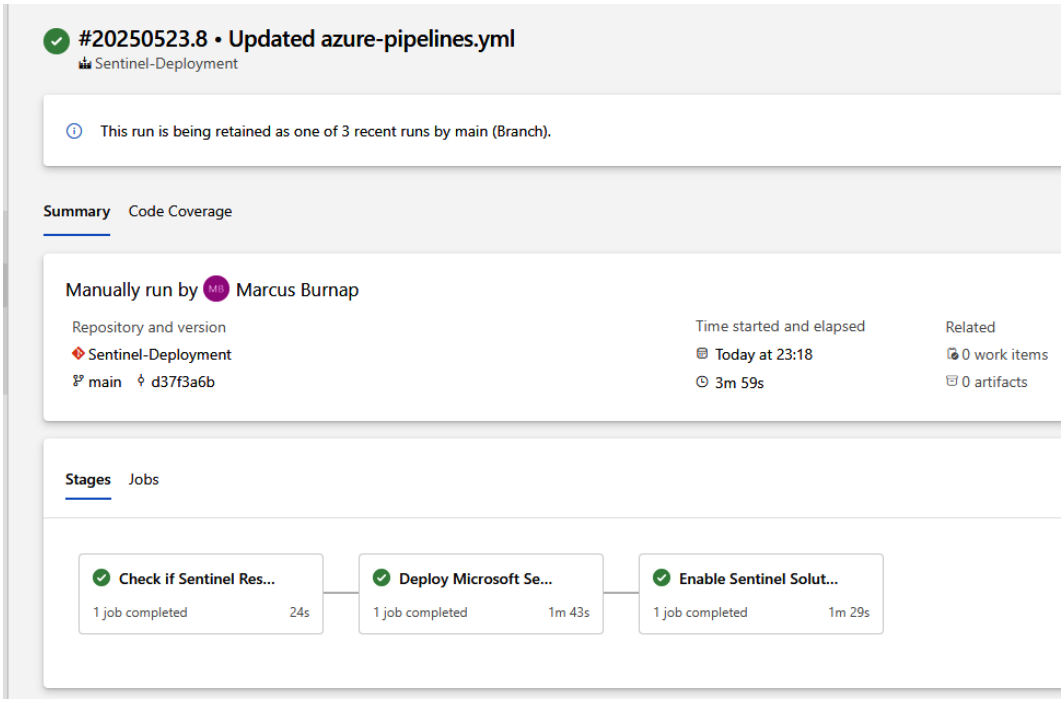


Figure 9.41: Successful pipeline deployment of each stage

You can now see the deployed content in the newly created resource group.

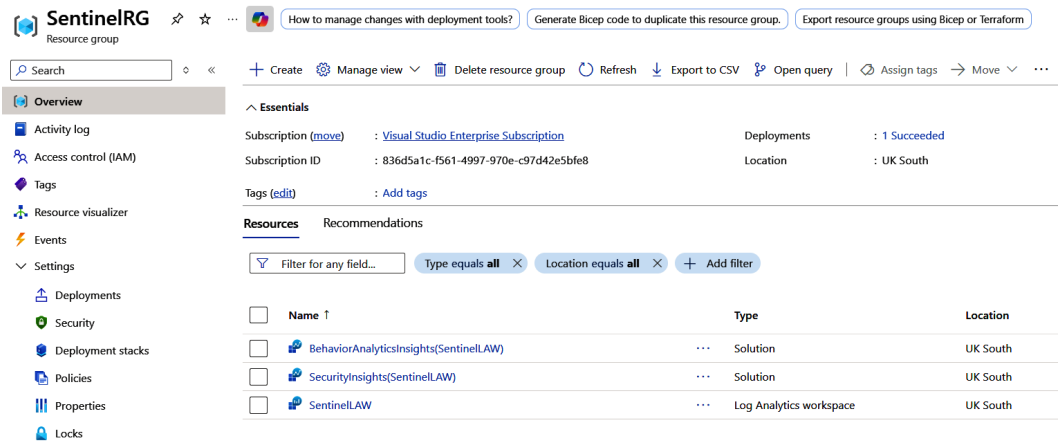


Figure 9.42: See the created Sentinel items

This completes the initial deployment stage. From here, you can iterate safely, refining the code and expanding the pipeline stages to validate changes and introduce additional content over time.

Understanding the pipeline workflow

The pipeline consists of three main stages:

- Stage 1 - `CheckExistingResources`: This stage performs a critical check to determine whether Sentinel resources already exist in the target environment. It uses Azure PowerShell to query for the Log Analytics workspace and sets a pipeline variable based on the result.
- Stage 2 - `DeployBicep`: The DeployBicep stage provisions the necessary infrastructure using Bicep templates. It only runs if the resources don't already exist, as determined by the previous stage. The Bicep deployment creates a Log Analytics workspace and Sentinel solution.
- Stage 3 - Enable Sentinel content via the content hub. The final stage deploys the Sentinel content and artifacts using our powerful PowerShell script. This stage runs regardless of whether the infrastructure was deployed in the previous stage, ensuring that content stays up to date even in existing environments.



Check this link for more information: `https://github.com/PacktPublishing/Microsoft-Sentinel-Blueprint/blob/main/Sentinel-Via-Bicep-main/Sentinel-Via-Bicep-main/Sentinel-Via-Bicep/Scripts/Set-SentinelContent.ps1`

The PowerShell script: Set SentinelContent.ps1

The heart of the solution is the `Set SentinelContent.ps1` script that handles the deployment of Microsoft Sentinel content. This script uses a unified testing framework to determine the status of various Sentinel resources and handles deployment, updates, and metadata association.

The `Set SentinelContent.ps1` script handles deploying Sentinel solutions from the content hub, configuring analytics rules based on specified severities, deploying associated workbooks, and properly linking all resources with metadata

Key features include the following:

- Unified status testing: Uses a consolidated function for checking resource status
- Intelligent update management: Skip or force updates based on requirements
- Professional error handling: Graceful handling of errors with detailed reports
- Comprehensive deployment: Handles solutions, rules, and workbooks in one script

So, there you have it – your Bicep deployment path is all set!

Deployment via Terraform

In the context of a large-scale Microsoft Sentinel deployment project, we are faced with the challenge of delivering a consistent, scalable rollout across multiple tenants, each with its own unique requirements and complexities. To address this, we have architected an automated deployment framework purpose-built for enterprise environments.

At the heart of this framework is Terraform, our chosen IaC solution. By embedding it directly into Azure DevOps pipelines, we have created a fully automated and repeatable deployment process. This integration allows us to standardize the rollout of Microsoft Sentinel and its foundational components across all customer instances, regardless of size or structure.

The power of combining Terraform's declarative configuration with Azure DevOps automation means we can accelerate deployment timelines, maintain strict consistency across environments, and reduce human error. This, in turn, strengthens the security posture of each tenant by ensuring that critical configurations are deployed uniformly and aligned to organizational standards.

It's worth noting that while Terraform provides broad coverage for many Azure resources, there are cases where certain Sentinel features or configurations are not natively supported by its modules. In those instances, we extend the automation by incorporating direct API calls by allowing us to bridge any gaps and preserve full end-to-end automation.

Azure DevOps pipeline integration

To bring the Terraform automation framework to life, we have built a fully integrated Azure DevOps pipeline to handle deployments across multiple tenants. While Azure DevOps is our platform of choice, the same design principles could easily be adapted for GitHub Actions or other CI/CD tools.

The pipeline is carefully structured with defined stages, triggers, and conditions to support robust, repeatable, and fault-tolerant deployments. Each stage, from code checkout and environment initialization to execution and post-deployment validation, is designed to ensure consistency and minimize the risk of human error.

State management strategy

A critical element of the deployment framework is the implementation of remote state management. For this, we use Azure storage accounts to hold the Terraform state files, ensuring that each tenant's deployment state is tracked independently and securely. This enables precise control over updates, change tracking, and rollback scenarios, especially important when dealing with multiple parallel environments.

Testing and validation

Before rolling the solution out into production environments, we conduct rigorous testing across dedicated test tenants. Every component of the automation, from the integrity of the Terraform scripts and pipeline mechanics to the reliability of state management, is validated. These tests are essential to ensure confidence in the automated rollout and to detect potential issues before they can impact live environments.

Terraform automation and deployment architecture for multi-tenant Sentinel environments

This section outlines a modular, secure, and scalable automation approach for deploying Azure Sentinel across multiple client tenants using Terraform and Azure DevOps. At its core, the solution automates the deployment of Microsoft Sentinel instances into Azure sub-tenants by leveraging parameterized Terraform scripts triggered through the central DevOps pipeline.

The architecture is built to support secure, scalable, and controlled deployments from a central home tenant, with careful consideration given to resource permissions, secure credential handling, and efficient state separation. This design not only simplifies management but also ensures a clear boundary of control and accountability across environments.

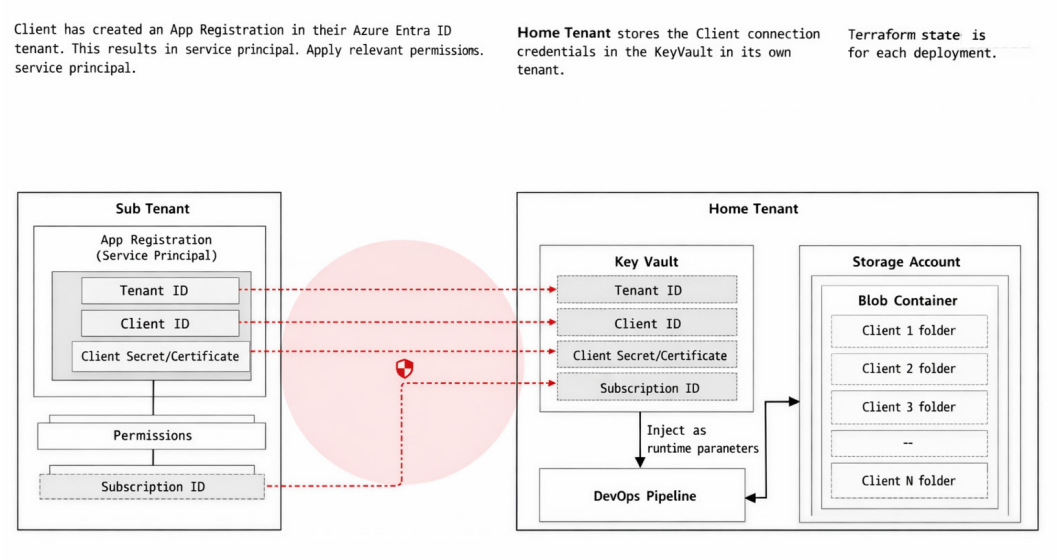


Figure 9.43: The Terraform MSSP deployment architecture

With cross-tenant access and identities in place, we will now focus on how deployments actually run from a single, controlled automation environment, into multiple client tenants, using a repeatable pipeline. This execution model is designed to give MSSPs and central SecOps teams consistent delivery, strong separation between clients, and secure handling of credentials and state.

Azure DevOps pipeline execution

This section will walk will walk you through the end-to-end pipeline architecture: where the automation lives, how Terraform is parameterized per client at runtime, how Azure Lighthouse enables delegated deployment without persistent admin access, and how Key Vault and Azure Storage are used to protect secrets and maintain reliable Terraform state. With that context set, we'll now run through the key Terraform elements and how they map to each stage of the pipeline.

1. Deployment environment (home tenant): The core automation environment is hosted in the home tenant. An Azure DevOps project serves as the central hub for managing Terraform code and pipelines. This setup ensures consistent and repeatable deployments across client Azure tenants.
2. Pipeline configuration: An Azure pipeline within this DevOps project orchestrates the deployment process using parameterized Terraform scripts. The pipeline enables modular and client-specific deployments by injecting variables dynamically at runtime.

3. Cross-tenant access with Azure Lighthouse: Azure Lighthouse is used to establish secure and trusted cross-tenant access. It allows the home tenant to manage and deploy resources within each client tenant, without requiring direct administrative access.
4. Client tenant configuration (Entra ID and service principal): Each client tenant has a dedicated app registration, known as the client service principal, created within Entra ID. This identity is assigned to the Contributor role in the client's Azure subscription, enabling Terraform to provision Sentinel-related resources securely.
5. Secure credential storage: Credentials for each client service principal are securely stored in the home tenant key vault. This enables safe retrieval during automated deployments without exposing secrets in code or pipelines.
6. Home tenant configuration (Entra ID and service principal): An additional home tenant service principal is created to access shared resources such as the Key Vault and Terraform state storage. It is granted a Key Vault Reader role to read stored credentials and the Storage Blob Data Contributor role to manage state files in the designated storage container.
7. These permissions enable the pipeline to authenticate and interact securely with both home and client tenants.
8. State management with Azure Storage: A dedicated Azure storage account in the home tenant is used to store Terraform state files, ensuring a secure and centralized state management approach.
    - Blob container: Stores isolated state files for each client
    - Access control: The home tenant service principal manages state access securely using its assigned role
9. Code and pipeline hosting (Azure DevOps or equivalent): This includes the following:
    - Git repository: Hosts the modular Terraform code base in the IaC or DevOps project
    - Pipeline setup: The Azure pipeline automates the end-to-end process using reusable modules and parameterized scripts
    - Automation structure: The code is organized for scalable deployment, with two top-level directories.

10. Code base structure: This structure includes the following:
    - clients folder: Contains YAML files, one per client, that define unique configuration values, including the following:
        - Client variables: Resource group names, environment tags, and identifiers
        - Backend variables: Subscription IDs and backend details for managing Terraform state in the blob container
    - resources folder: Contains the main Terraform logic. Key files include the following:
        - `variables.tf`: Defines all input variables, such as subscription ID, client ID, tenant ID, and secrets
        - `providers.tf`: Configures the Azure provider dynamically using runtime variables injected by the pipeline
        - `backend.tf`: Sets up remote state storage using dynamically assigned container and key vault values
        - `outputs.tf`: Exposes outputs such as resource IDs and connection strings for downstream use
        - `main.tf`: The central file orchestrating deployment by calling reusable modules
    - `modules/` folder includes the following:
        - `log_analytics`: Provisions Log Analytics workspaces
        - sentinel: Deploys Azure Sentinel configurations
11. Pipeline specification (`azure-pipelines.yml`): This file defines the CI/CD workflow with the following:
    - Variable injection: Pulls client-specific YAML values into each pipeline run
    - State management: Dynamically assigns storage container settings per client
    - Terraform execution: Runs init, plan, and apply commands in sequence, ensuring controlled and validated deployments

This structure enables secure, repeatable, and modular deployments of Microsoft Sentinel across a multi-tenant environment, streamlining automation for MSSPs and enterprise-scale security teams.

Azure DevOps: project setup

Create an Azure DevOps organization if you don't already have one. Create a new project within DevOps and push the following code nto the repo of the home tenant project.

```
Terraform_Sentinel
├─ clients
│  └─ <client-name>
├─ resources
│  └─ modules
│     ├─ log_analytics
│     │  ├─ main.tf
│     │  ├─ outputs.tf
│     │  └─ variables.tf
│     └─ sentinel
│        ├─ main.tf
│        ├─ outputs.tf
│        └─ variables.tf
├─ backend.tf
├─ main.tf
├─ outputs.tf
├─ providers.tf
├─ README
├─ terraform.tfvars
├─ tfplan
├─ variables.tf
├─ azure-pipelines.yml
└─ README.md
```

Figure 9.44: Example repo setup in Azure DevOps as per the architecture model explained previously

Congrats, you have your base project set up. The next section covers the minimum secure foundation required to deploy Microsoft Sentinel via Terraform at scale. The goal is to avoid hard-coded credentials, ensure deployments are repeatable, and keep each client isolated through clean configuration boundaries.

As you work through the chapter, you'll notice a consistent pattern: we first establish secure identity and storage in the home tenant (because that's where automation runs), then we prepare each client tenant with the least privilege access Terraform needs to deploy Sentinel resources.

Deploying Sentinel with Terraform: Key Vault and sub-tenant setup

Let's start by creating a dedicated Azure Key Vault for this project:

1. To create a new Azure key vault, select your subscription and add a new resource group.
2. Enter a key vault name, and select a region and pricing tier.
3. Select **Review | Create** when finished.

Your pipeline needs to retrieve client credentials securely at runtime. Key Vault prevents secrets from being exposed in code, variable groups, or pipeline logs.

Once the Azure key vault is created, the next step is to create a service principal:

1. In Microsoft Entra ID, go to **App Registrations** and register a new app registration.
2. Note down the client ID and tenant ID.
3. Click on **Certificates & secrets** | **New client secret**.
4. Enter a name and click **Add**.
5. Note down the secret value (this will not be shown again).

The pipeline needs an identity to read secrets from Key Vault and access the Terraform state storage account. This service principal represents your automation worker identity.

Now, you must configure Key Vault access permissions:

1. In the Azure Portal, go to the Key Vault created earlier.
2. Navigate to **Access Policies** and click **Add Access Policy**:
3. Add **Get** and **List** under **Secret permissions**.
4. Under **Select principal**, choose the service principal created earlier.
5. Click **Save**.



If your organization uses Azure RBAC for Key Vault, use the Key Vault Access control (IAM) blade instead and assign a suitable role (e.g., Key Vault Secrets User). Keep the rationale the same: least privilege to read secrets.

Adding backend secrets

Follow these steps to add backend secrets:

1. Open the Key Vault and navigate to **Objects** | **Secrets** | **Generate/Import**.
2. Add the following secrets using the values captured during service principal creation:
    - `backend-client-id`
    - `backend-client-secret`
    - `backend-key (same as backend-client-secret)`
    - `backend-subscription-id`
    - `backend-tenant-id`

Terraform needs backend configuration values and authentication details to store state centrally and securely. Storing these as secrets keeps your backend consistent and reduces accidental leakage.

Creating a new service connection in Azure DevOps

The service connection is how Azure DevOps securely authenticates to Azure during pipeline execution. It provides governance, traceability, and permission boundaries at the DevOps layer. To create a new service connection, follow these steps:

1. Sign in to your Azure DevOps organization and navigate to your project and click **Project settings** (bottom left).
2. Select **Service connections**.
3. If this is your first connection, click **Create service connection**; otherwise, click **new service connection**.
4. Select **Azure Resource Manager** | **Next**.
5. Choose Service principal (manual) | **Next**.
6. Select **Subscription for Scope Level** and enter the following:
    - Service Principal Id: `App ID`
    - Service Principal key: `Secret`
    - Tenant ID
7. Provide a name for the service connection.
8. Check the box **Grant access permission to all pipelines**.
9. Click **Verify and Save**.

Creating a storage account

Remote state enables consistent deployments and prevents local state drift. It also supports collaboration and protects against accidental overwrites. You can use an existing account or create a new storage account. Then, assign the **Storage Blob Data Contributor** IAM role to the service principal.

Creating a client blob container in the storage account

To create a client blob container , follow these steps:

1. Log in to the Azure portal and open the storage account.
2. Go to **Storage browser | Blob containers**.
3. Click **Add container**.
4. Enter a name, Set **Public access level** to `Private`.
5. Click **Create** and open the container.

Isolating state per client reduces the blast radius and prevents cross-client state collisions. It also makes destroy/rebuild operations safer and more predictable.

Sentinel Terraform sub-tenant setup

We will now look at the outlines for sub-tenant (client) requirements using the Sentinel deployment pipeline. Parameterizing per tenant keeps the code reusable. A single pipeline can deploy to many clients by switching variables rather than duplicating code.

Creating a client blob container

Because the state configuration lives in the home tenant, we must manually create a new storage location.

1. Log in to the Azure portal and open the storage account.
2. Navigate to **Blob containers**.
3. Click **Add container**, enter the client name as the container name.
4. Set **Public access level** to **Private**.
5. Click **Create** and open the container.

Preparing a client variables file

Each client has its own identity and subscription context. Storing secrets per client in Key Vault allows secure, automated retrieval during deployments without exposing them in repo files.

Create a new `.yaml` file in the client folder (use `swat.yaml` as a template) and populate the file with the following:

- `client_name`
- `resource_group_name`
- `location`
- `log_analytics_workspace_name`
- `retention_in_days`
- Add blob storage values:
    - `backend_container_name`
    - `backend_resource_group_name`
    - `backend_storage_account_name`

  Now, add a new blob container named with the client identifier.

Using the client values, create the following Key Vault secrets (prefix with the client identifier) to add client secrets:

- `client-id`
- `client-secret`
- `subscription-id`
- `tenant-id`

Executing the pipeline

It's time to execute the pipeline. This is where the model becomes operational. Your repo content, secrets, and state configuration are combined into a controlled deployment with an audit trail.

Follow these steps to execute the pipeline:

1. Select the pipeline and click **Run pipeline**.
2. Enter the client identifier.
3. Select the action (**Build** or **Destroy**).
    - Default: Build
4. Select Log Level.
    - Default: Error
5. Click Run and track the execution.



Lessons from the field

To ensure scalable, secure, and maintainable deployments, follow these best practices when automating Microsoft Sentinel with Terraform:

- Use reusable modules to standardize common Sentinel configurations across environments.
- Implement state management by storing Terraform state files in an Azure storage account with state locking enabled to prevent conflicts.
- For secret management, avoid embedding credentials in pipeline code by leveraging Azure Key Vault or Workload Identity Federation.
- Maintain version control by storing ARM templates, KQL queries, and analytics rule configurations in a source-controlled repository.
- Finally, integrate testing into your pipeline using terraform validate and automated tests to catch errors early and ensure deployment reliability.

CI/CD automation for Azure Sentinel significantly enhances your security posture by enabling consistent, auditable, and scalable deployments. By leveraging Terraform and Azure DevOps, security teams can manage infrastructure, deploy rules, onboard connectors, and configure workbooks with confidence and speed. This implementation paves the way for future enhancements such as deploying playbooks (Logic Apps), integrating with Defender for Cloud, and enabling Sentinel as Code using Azure Verified Modules.

Summary

In this chapter, we discussed how organizations industrialized their Sentinel operations by adopting IaC and CI/CD pipelines. You were also introduced to reusable repository structures in Azure DevOps or GitHub, which allow content such as analytics rules, workbooks, and playbooks to be deployed consistently across workspaces. The chapter demonstrated how to build modular pipelines with per-tenant configuration files, solution pickers, and YAML templates, and showed how smart deployment logic reduces duplication and enhances governance.

It also introduced Workspace Manager (Preview) as a UI-driven method for deploying content across multiple workspaces, which is ideal for organizations not yet ready for full automation. By adopting these strategies, MSSPs and large enterprises were able to streamline onboarding, reduce errors, and maintain a consistent security baseline across all environments.

With automation in place and Sentinel deployment fully streamlined, in the next and final chapter, we will shift our focus to the future of security operations. We will explore Microsoft Security Copilot, showing how AI and large language models are transforming how analysts investigate, reason, and respond to threats.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

Note: *Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



10

Activate, Configure, Deploy: Your First Steps with Security Copilot

In this chapter, we will establish the prerequisites, architectural concepts, and access models needed before deployment. This chapter will help you build foundational knowledge of how Security Copilot works, what resources are required, and how to approach security and access governance.

We will cover the following topics:

- Introduction to Microsoft Security Copilot and its capabilities
- Foundational requirements: Azure subscription and SCUs
- Authentication and delegated access model
- Onboarding process: getting Security Copilot up and running
- Key topics around Microsoft Security Copilot and AI integration
- Future outlook: Microsoft Sentinel and Defender XDR

By the end of this chapter, you will be prepared to design and plan a secure, compliant, and scalable Security Copilot deployment before touching any configuration.

Technical requirements

To test Security Copilot, you will need an active M365 tenant and an Azure subscription. We would highly recommend looking at the latest licensing guidance. As in December 2025, Microsoft offers free capacity with some paying licensee types.

Introduction to Microsoft Security Copilot and its capabilities

Think of Microsoft Security Copilot as your AI-powered security partner. Security Copilot combines the strengths of advanced **large language models (LLMs)**, such as GPT-4, with Microsoft's repertoire of vast threat intelligence. It actively learns, connects the dots, reasons through complex threats, and helps you take real, actionable steps. You interact with it using natural language, just like you're having a conversation. As of 2025, Microsoft processes over 84 trillion security signals daily, encompassing data from email, identity, endpoints, cloud services, and more.

Security Copilot brings together a set of features that transform how modern security teams operate. With natural language queries, you can ask questions in plain English to analyze incidents, summarize alerts, or receive step-by-step remediation guidance. This eliminates the need to learn a specialized query language. It leverages integrated threat intelligence from Microsoft Defender XDR, Sentinel, and global threat signals, providing in-depth, contextual insights rather than isolated data points.

Promptbooks, which are curated collections of predefined prompts and workflows, guide Security Copilot through common security scenarios such as incident triage, threat hunting, or compliance checks. They act as reusable templates for routine security tasks, helping standardize operations, save time, and ensure consistency across the SOC. Thanks to conversational memory, Copilot remembers interactions during a session, allowing users to refine queries and dig deeper without starting over. And with **role-based access control (RBAC)** built in, sensitive information is only visible to authorized personnel, keeping privacy and data security firmly in check.

With promptbooks establishing consistent, repeatable workflows for Security Copilot, we need to ensure that the underlying environment is properly prepared. In the following section, we will discuss the foundational requirements needed to enable and scale Security Copilot effectively.

Foundational requirements: Azure subscription and SCUs

Firstly, you'll need an active Azure subscription to use Security Copilot.



Don't have one? No problem. Head over to `https://portal.azure.com/` and spin up a free account to get started.

Then, you need computing power to get the engine running. In Security Copilot, this is called **Security Compute Units (SCUs)**.

Understanding SCUs: provisioned versus overage

Once you have the subscription up and running, you will need SCUs. Security Copilot runs on them. Think of these as the fuel that powers your Copilot, helping ensure consistent performance as demand grows.

There are two types of SCUs to keep in mind:

- **Provisioned capacity**: These are predictable, always-on resources that are billed hourly. They're ideal for steady workloads.
- **Overage capacity**: These are pay-as-you-go resources that kick in during traffic spikes. They're great for flexibility when things get busy.

Keep an eye on consumption using Copilot's built-in monitoring dashboard. It helps you make smart decisions about when to scale up (or down) based on real demand.



SCUs for Security Copilot are purpose-built. You can't reuse ones already allocated to other services, such as Microsoft Purview. Each service needs its own dedicated SCU pool.

With an understanding of how SCUs are provisioned and consumed, we will now focus on how access is securely established. The following section explains the authentication and delegated access model, outlining how identities, permissions, and trust are managed across environments.

Authentication and delegated access model

Before using Microsoft Security Copilot, we need to understand how authentication and access control are structured. Security Copilot operates using delegated, "*on-behalf-of*" authentication. This means that users interact through Microsoft plugins with access limited strictly by the roles they've been assigned. Getting these permissions right from the outset is essential to ensuring a secure and functional deployment.

Once authentication and access control are established, we will shift our focus to platform governance, including RBAC, plugins, and collaboration settings. These will all affect the overall posture of the tool.

Role-based access in Security Copilot

To use Security Copilot effectively, your team needs a combination of Microsoft Entra roles, Azure RBAC, and Copilot-specific roles.

The following are the types of roles involved:

- **Security Copilot roles**: These roles are specific to the Copilot platform itself and determine access to its features.
    - **Copilot owner**: Grants full administrative access to manage configurations, install plugins, and handle all admin-level tasks.
    - **Copilot contributor**: Grants permission to use the platform, run promptbooks, and create sessions, but not to manage system-wide settings.
- **Microsoft Entra roles**: These govern access to identity and compliance services across Microsoft 365. Roles such as Security Administrator and Global Administrator can automatically elevate users to Copilot owner status.
- **Azure RBAC**: This controls access to Azure resources, including Microsoft Sentinel, SCUs, and other connected services.



Lessons from the field

Most organizations benefit from using the recommended Microsoft Security roles group. This group provides a secure and scalable method for assigning access across services such as Microsoft Defender, Sentinel, and Purview.

Avoid assigning Copilot access to the **Everyone** group, as this can result in users encountering features they are not authorized to use, leading to confusion and potential security risks.

When possible, use security groups instead of assigning roles to individuals. This approach simplifies role management, especially as the team scales.

Plugin permissions and the least privilege model

Security Copilot is deeply integrated with the Microsoft ecosystem through plugins, but it adheres to a strict least privilege model. Even if a user is a Copilot contributor, they will not be able to access plugin data unless they have the appropriate service-specific RBAC roles.

To access plugin data from critical Microsoft security tools, ensure the following roles are properly assigned:

- **Microsoft Sentinel**: Reader or contributor roles
- **Microsoft Intune**: Appropriate device or security roles
- **Microsoft Defender XDR**: Unified RBAC roles configured for endpoint or identity access

Each of these services has its own RBAC structure. Be sure the correct roles are assigned in both Azure and Microsoft Entra to ensure full integration with Copilot.



Lessons from the field

Advanced features such as embedded investigations and data enrichment require additional permissions or licenses. These are layered on top of basic Copilot contributor access and may vary depending on the plugin or feature being used.

Cross-tenant collaboration and shared sessions

Security Copilot is built with collaboration in mind, including scenarios that span across multiple Microsoft 365 tenants. Within a single organization, Copilot contributors can share session links with colleagues, and recipients do not require individual plugin permissions to view the shared content. This streamlines internal collaboration and accelerates review processes.

For multi-tenant environments, Security Copilot supports a range of Microsoft collaboration mechanisms, including Azure B2B, guest accounts, GDAP, and Azure Lighthouse. These capabilities enable secure cross-tenant access while maintaining role-based control and visibility.

Now that the foundational concepts are in place, let's walk through the hands-on process of deploying and configuring Microsoft Security Copilot in your environment.

Onboarding process: getting Security Copilot up and running

Getting started with Microsoft Security Copilot is refreshingly straightforward. Once your Azure subscription is ready, onboarding is just a few clicks away. Here's a step-by-step guide to help you provision capacity and configure your environment to begin using Security Copilot efficiently.

Step 1: Provisioning your capacity

Here, we will look at the two options you have to provision SCUs. While both work, we recommend starting through the **Security Copilot portal** for the smoothest experience.

Option 1: Provisioning via the Security Copilot portal

1. Navigate to the Security Copilot portal (`https://securitycopilot.microsoft.com`) and sign in. You will be presented with the following screen:

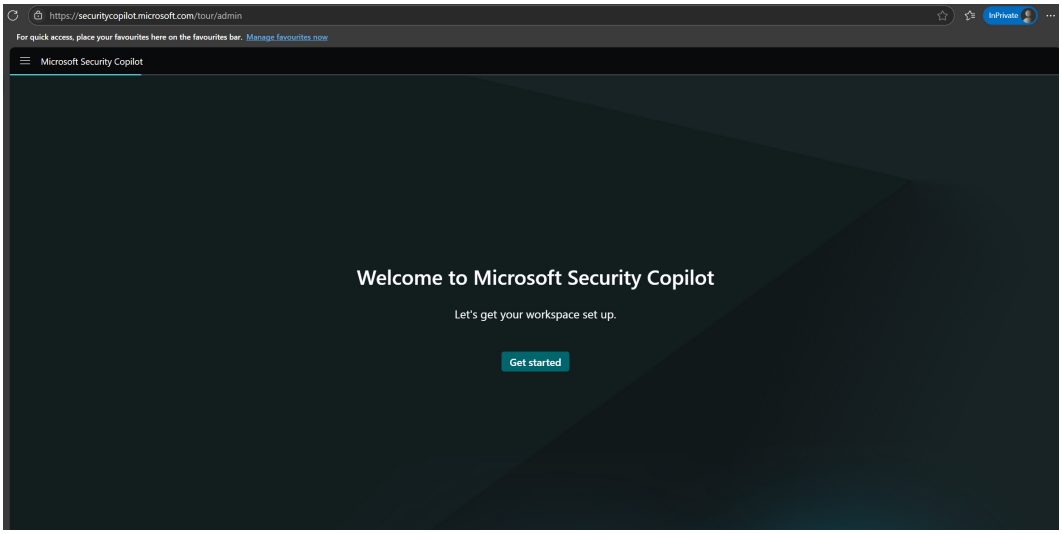


Figure 10.1: Navigate to the Security Copilot portal

2. Get started by setting up workspace information. Provide basic workspace details, including your name and the data storage location (e.g., **United Kingdom**) in the required fields. Click **Continue** to proceed.

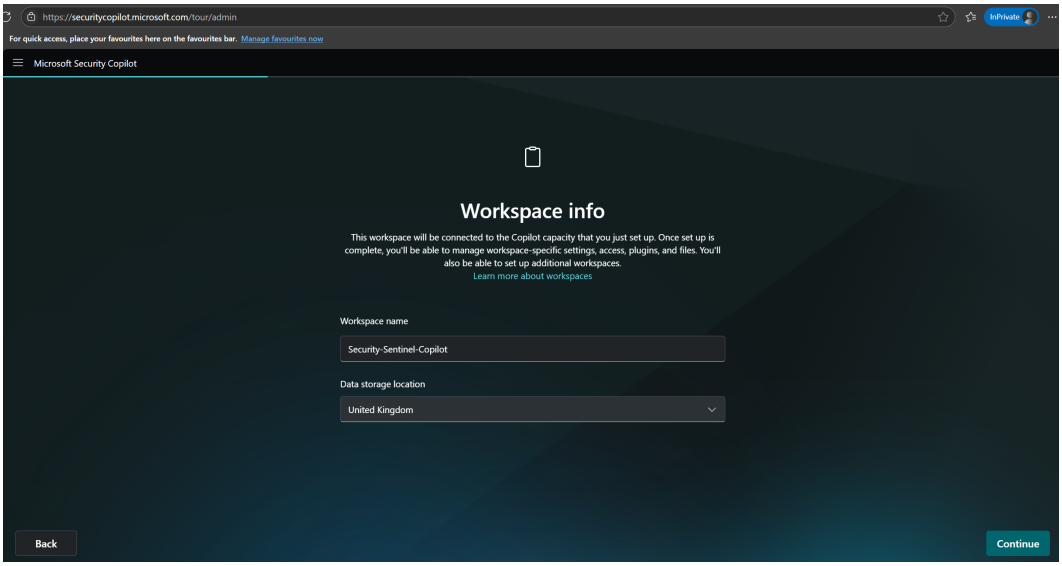


Figure 10.2: Enable Security Copilot settings

3. To configure security capacity, follow these steps:
    1. Select your Azure subscription where costs and resource management will occur.
    2. Choose an existing resource group or create a new one.
    3. Set a capacity name and prompt evaluation location.



If the selected region is experiencing high load, Copilot can evaluate prompts in other regions to improve performance.

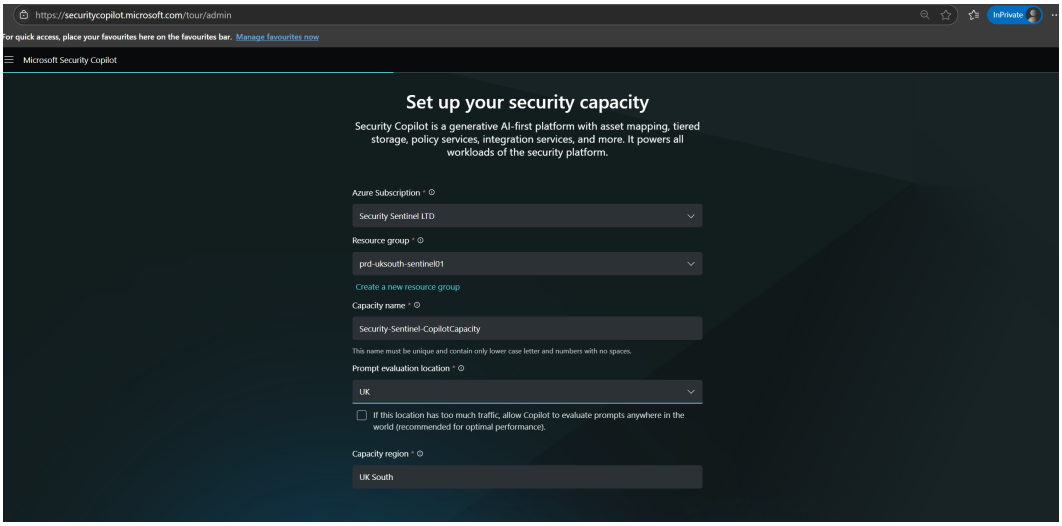


Figure 10.3: Enable Security Copilot capacity

Within this same screen, define how many SCUs you need, either provision, always-on, billed hourly, or overage (pay-as-you-go) for traffic spikes.

4. Review the cost estimate, accept the terms, and click on **Provision**.
5. **Optional**: You can choose whether to share data gathered from your organization's use of Microsoft Security Copilot, including user prompts, the security information that's accessed, and Copilot's responses, with Microsoft. You can change these settings at any time.

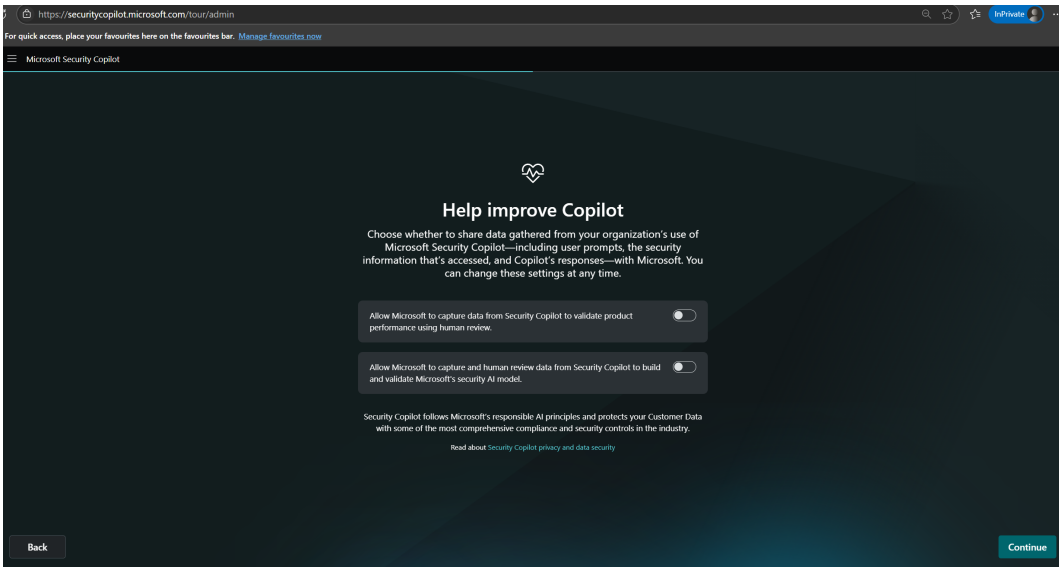


Figure 10.4: Enable Security Copilot – extended settings

6. Once created, select the new capacity to activate it and click **Continue**.

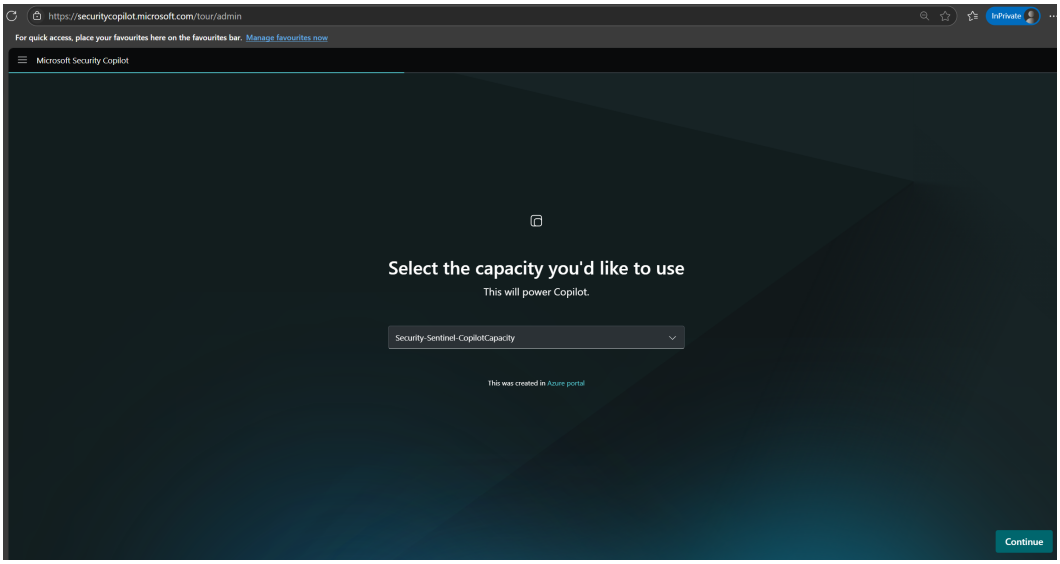


Figure 10.5: Enable Security Copilot – select capacity

This is what the capacity will look like in the Azure portal:

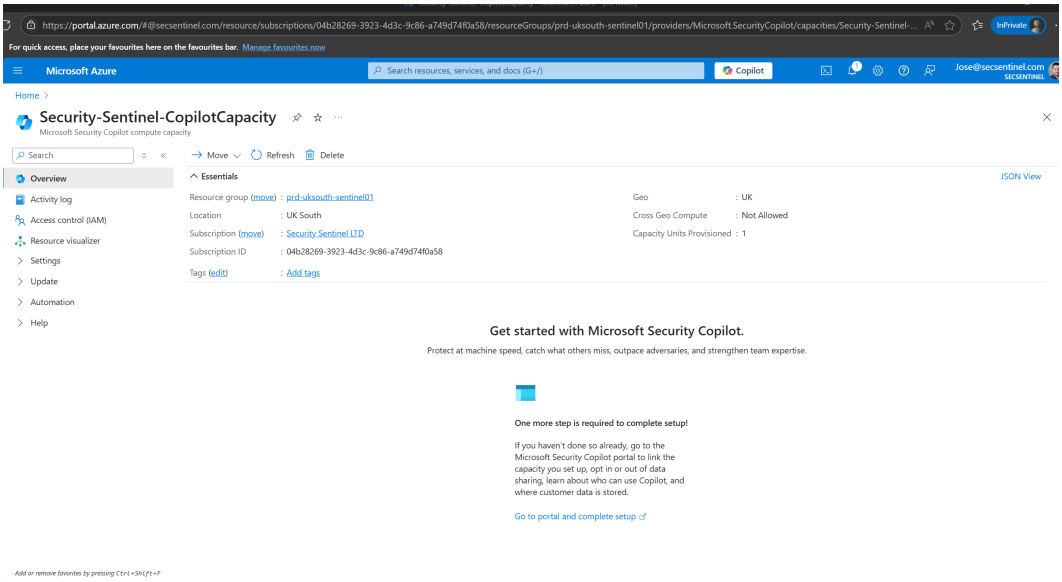


Figure 10.6: Enable Security Copilot – set up capacity in Azure

Option 2: Using the Azure portal

1. Log in to the Azure portal and search for the **Security Copilot** service in the search bar.

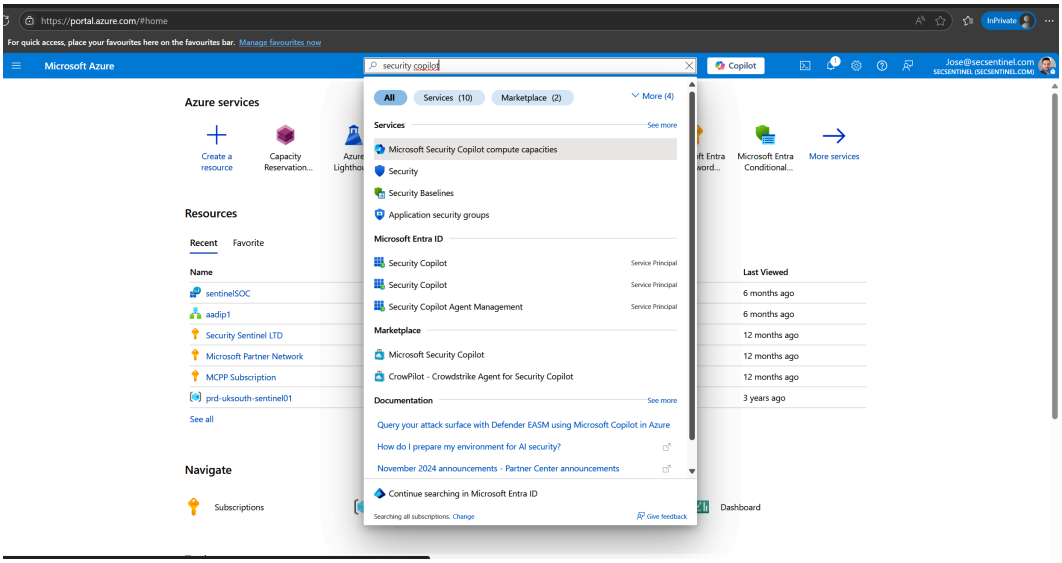


Figure 10.7: Search in Azure for Security Copilot

2. Click on **Create** to create a new capacity resource:

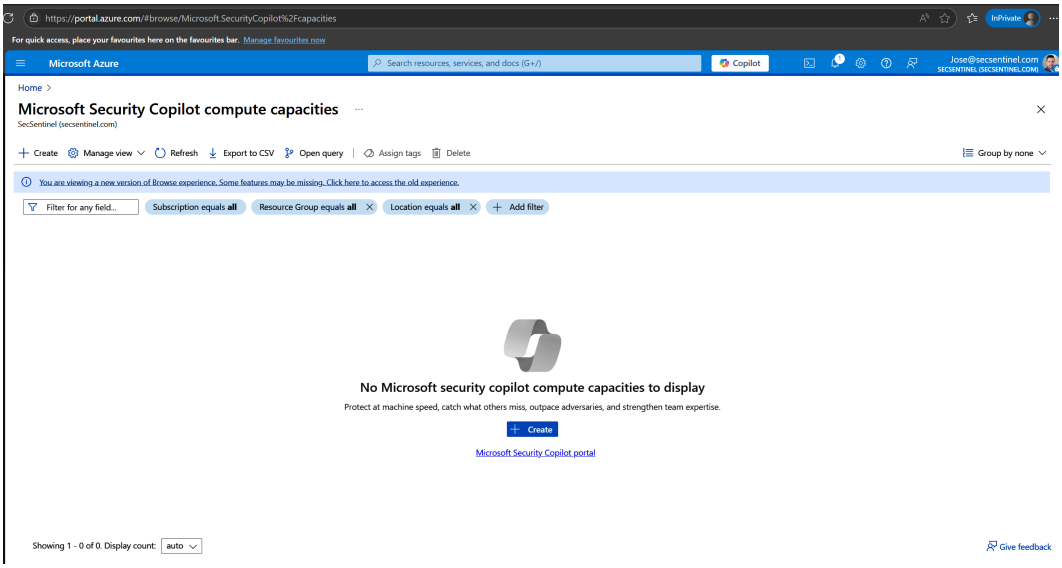


Figure 10.8: Create capacity in Azure

3. Configure the capacity – the same process as you would have done within the Security Copilot portal:

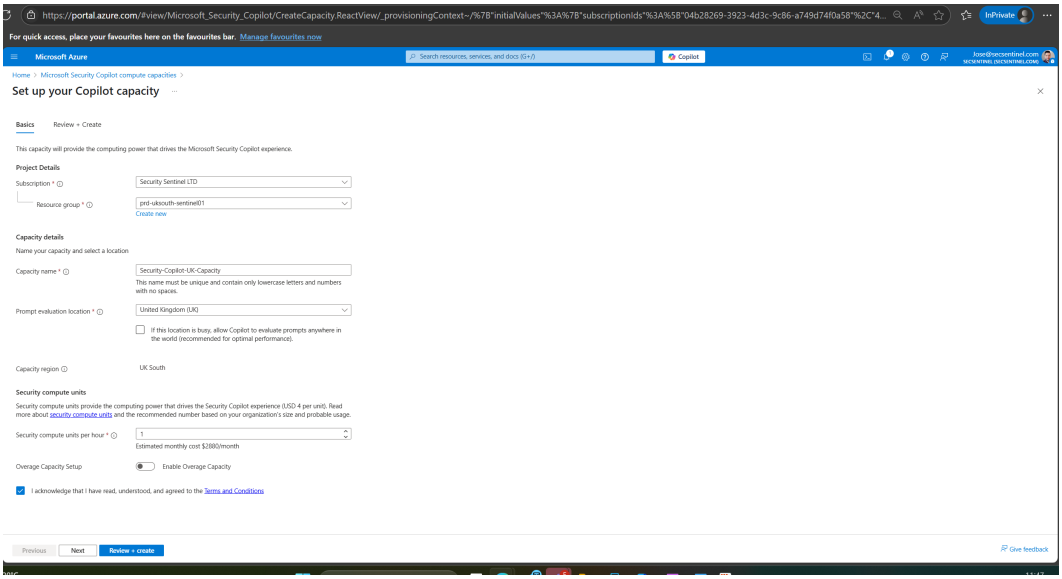


Figure 10.9: Set up capacity in Azure – settings

4. Review the configuration and finalize the setup.

   Congratulations, you have now successfully deployed an SCU.

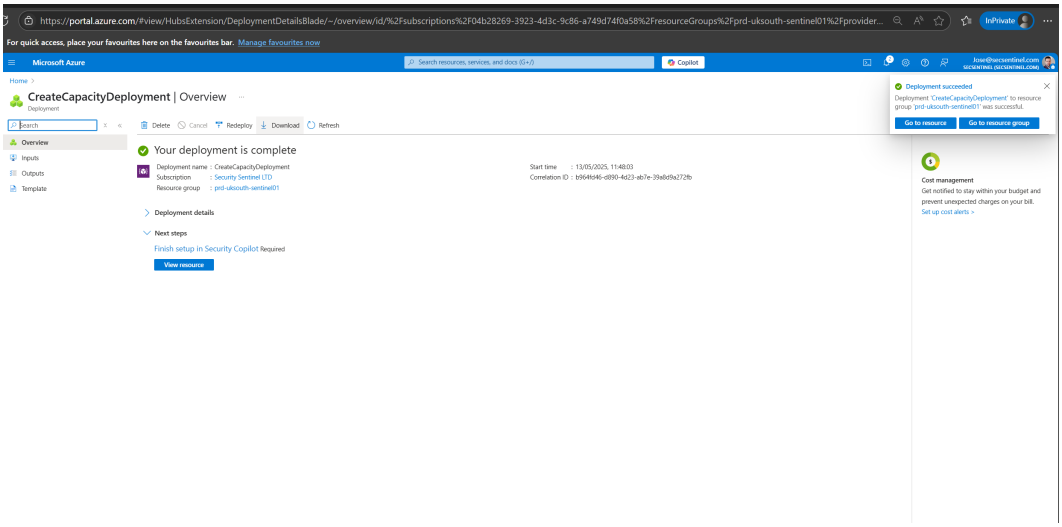


Figure 10.10: Capacity deployment completed

5. Now, if you go back to the Security Copilot portal, you will be able to see your capacity, as shown here:

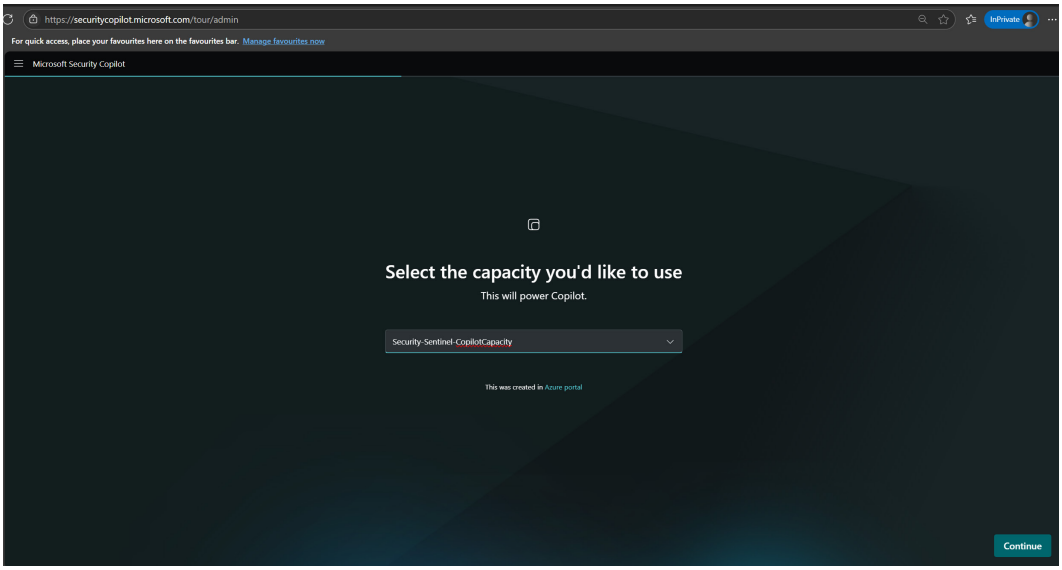


Figure 10.11: Security Copilot portal – select your capacity

With SCU capacity in place, it is time to configure your environment.

Step 2: Configuring your default environment

Microsoft Security Copilot can integrate with licensed Microsoft 365 and security products, allowing users to query data directly. This requires Copilot to copy, process, and store that data. If you opt out of data access in the owner settings, integration with services such as Microsoft Purview won't be available.

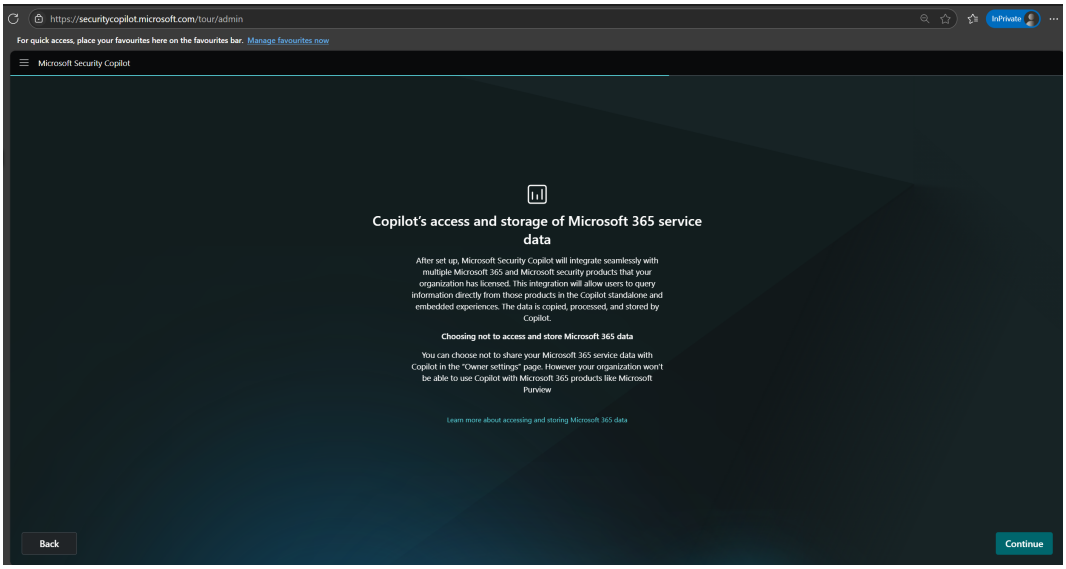


Figure 10.12: Security Copilot portal – accept terms and continue

1. Switch on the toggle available against **Allow Microsoft Purview to access...** to set up what Copilot will track, including admin actions, user activity, and system responses.

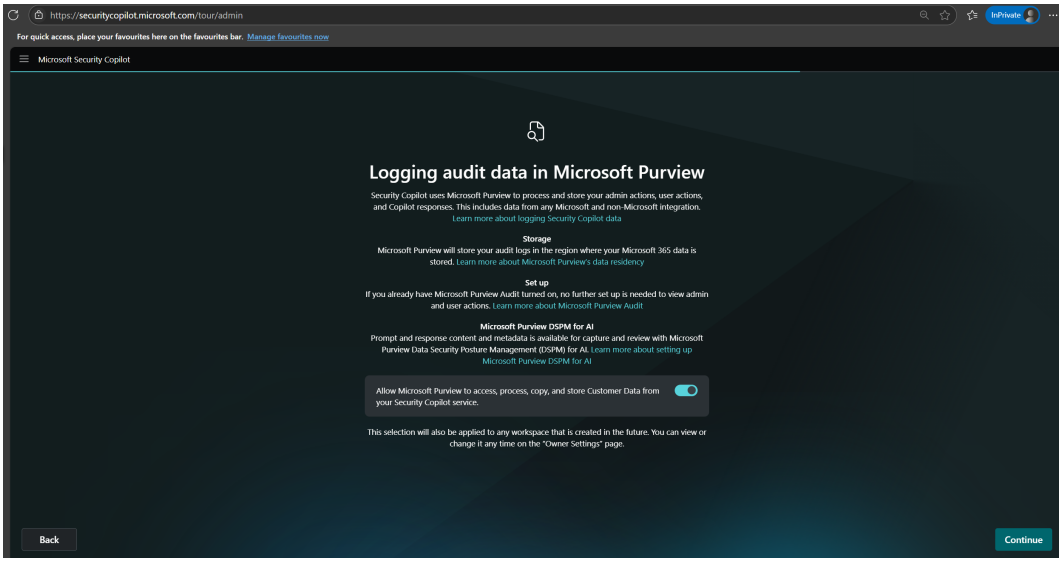


Figure 10.13: For auditing, allow Purview access and continue

2. You will see the built-in recommended roles to ease the management of Security Copilot operations. Select one to assign your roles.

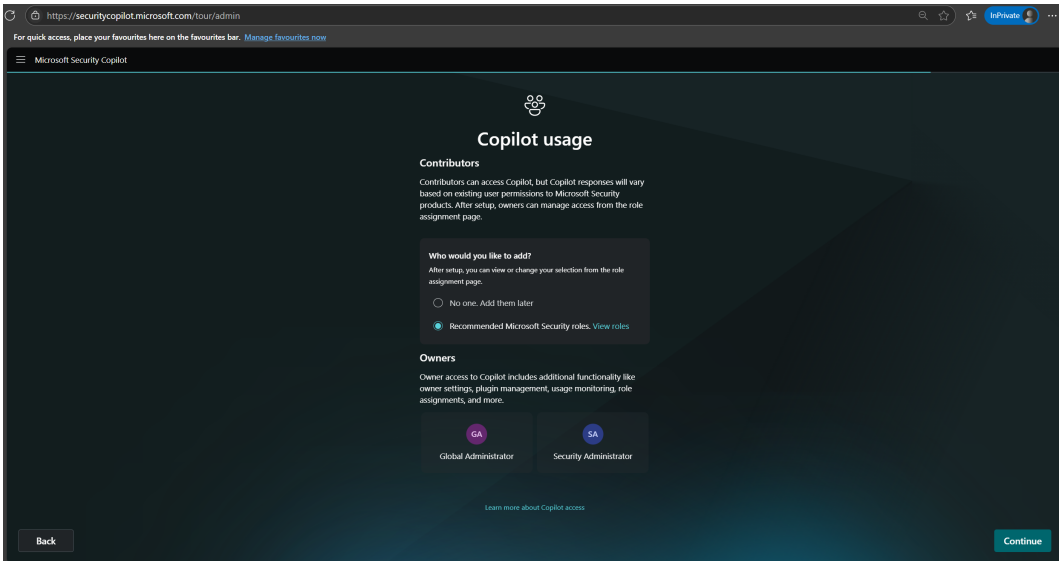


Figure 10.14: Security Copilot portal – assign roles

These are just a selection of the available roles. Our recommendation is for your organization to always align with zero trust and apply the least privilege principle. To start with, use the default and come back to these via the settings in the bottom left-hand menu.

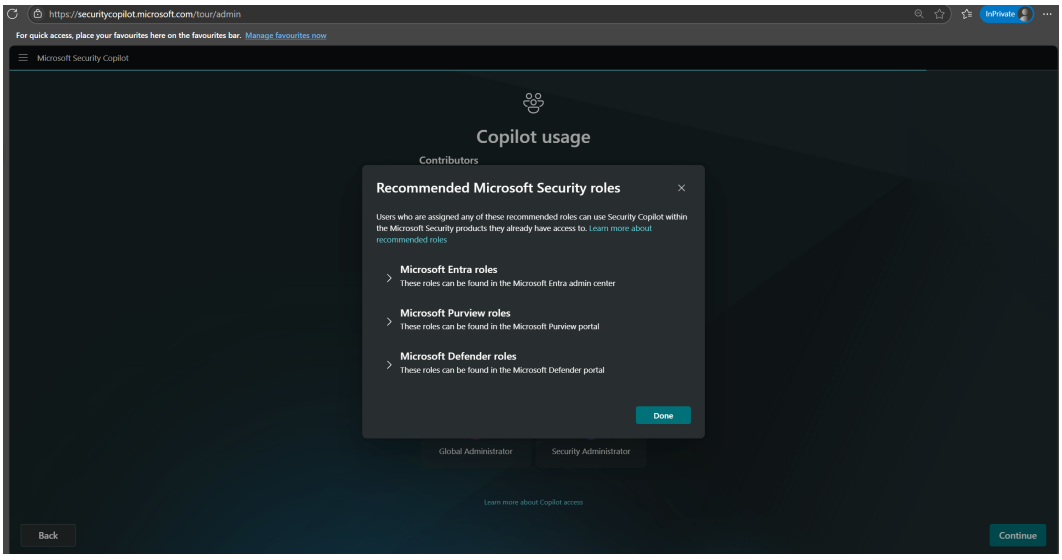


Figure 10.15: Security Copilot – recommended roles for admin management

3. Confirm everything and finalize the setup.

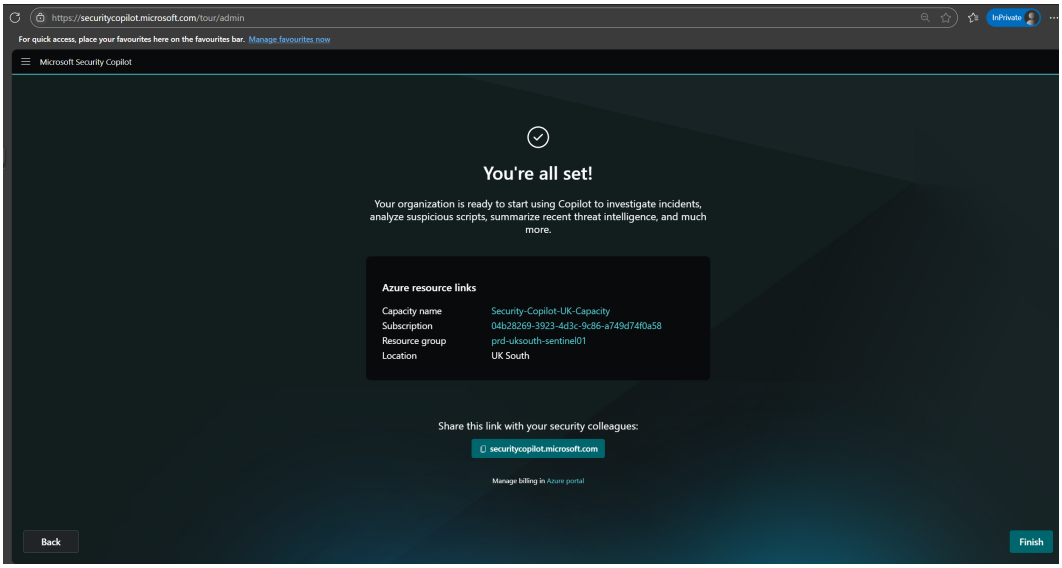


Figure 10.16: Security Copilot – setup complete

4. Test the configuration – try a prompt to ensure your deployment is up and running.

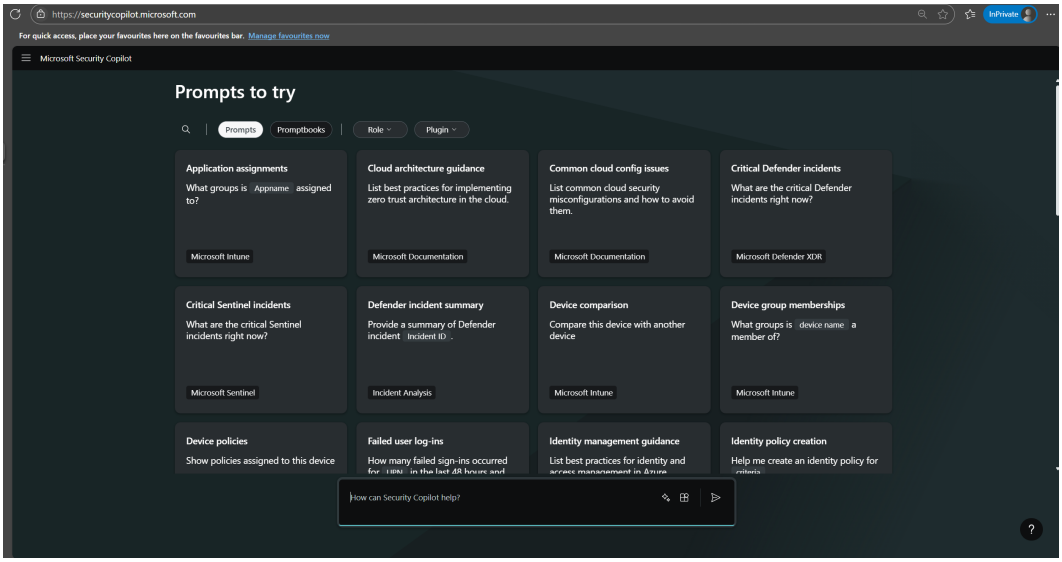


Figure 10.17: Security Copilot portal – built-in prompts

5. After testing a prompt, you can receive data from the executed query, explore it further, and view the completed actions.

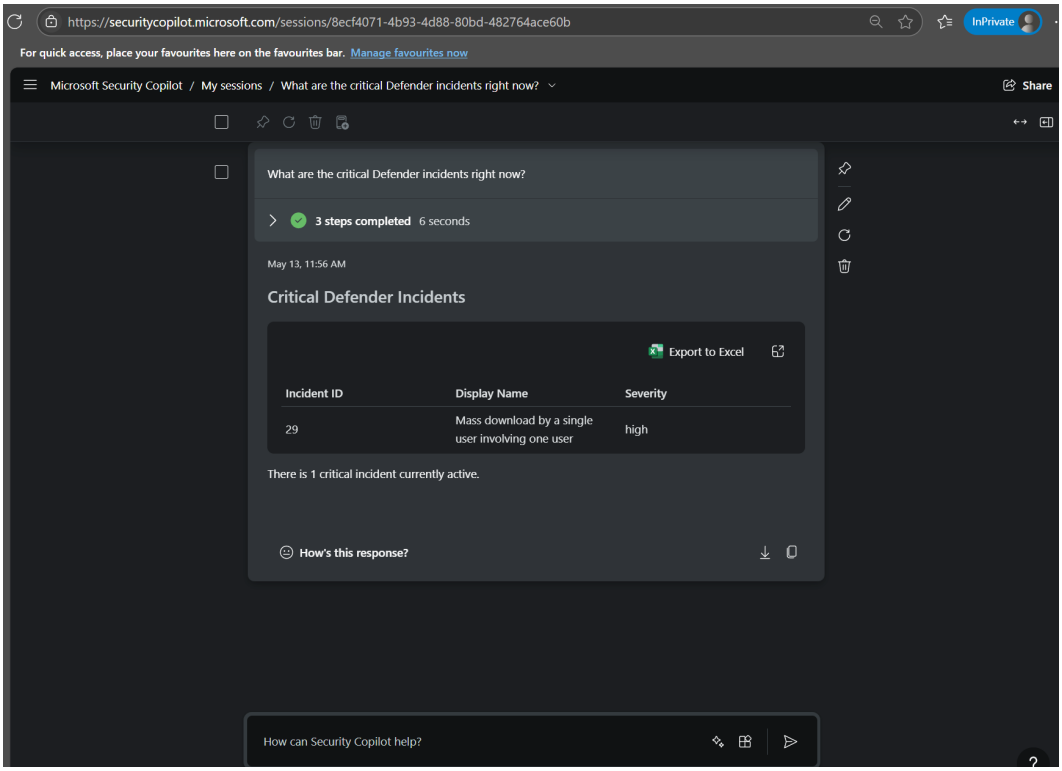


Figure 10.18: Security Copilot portal – prompt results

Offboarding: if you need to step away

If you ever decide to stop using Security Copilot, you'll need to delete the provisioned capacity. Follow these steps to do that:

1. Head to the **Owner** settings or the **Usage monitoring** dashboard in Security Copilot. Here, you can scale or reduce your capacity once per hour by clicking on **Change units**.

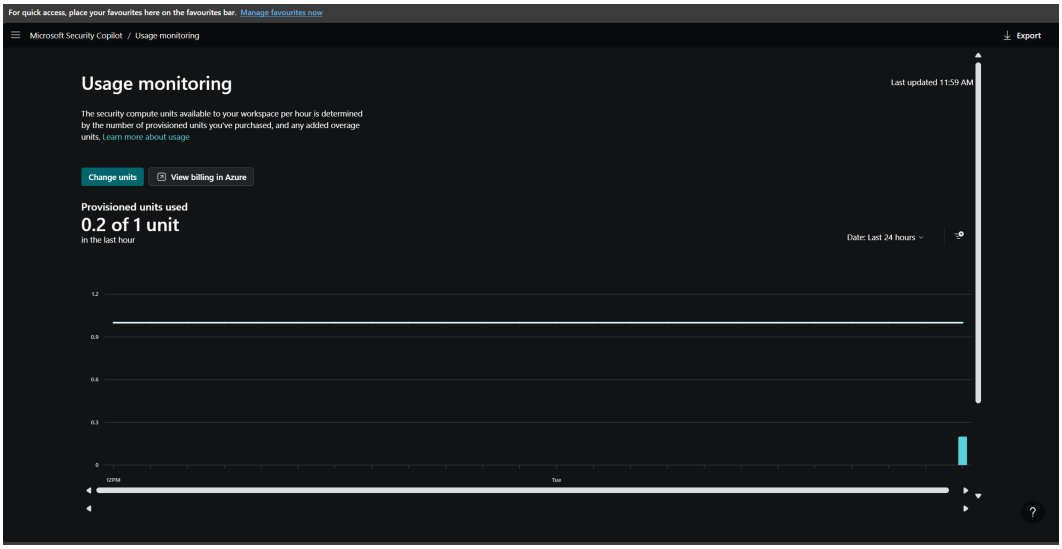


Figure 10.19: Security Copilot portal – usage dashboard

2. Choose the capacity you would like to remove. Then, click on the ellipsis icon and click on **Delete the capacity**. Note that you cannot undo this action.

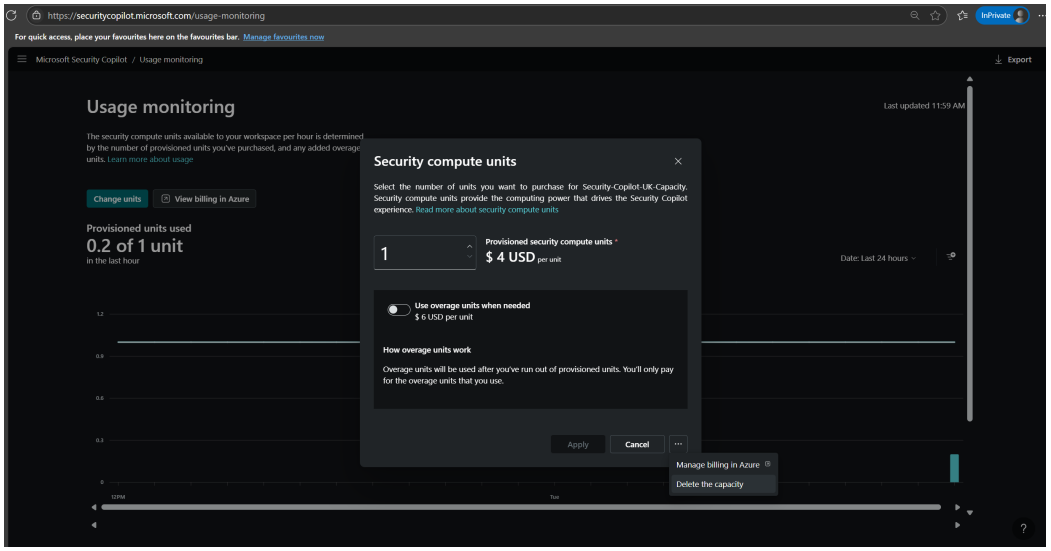


Figure 10.20: Security Copilot portal – offboarding

Now, let's look into some key topics around Microsoft Security and how it integrates with Security Copilot.

Key topics around Microsoft Security Copilot and AI integration

Microsoft is introducing several new capabilities into its ecosystem that will directly impact MSSPs, developers, and enterprise customers alike. In this section, we will discuss the key themes and what they mean for partners and customers.



Disclaimer

The following content is conceptual and reflects our current (2025) understanding and interpretation of where Microsoft's security products, such as Microsoft Sentinel, Defender XDR, and Security Copilot, are heading. While based on existing product direction, roadmap signals, and industry trends, the features and capabilities discussed may change over time and should not be taken as official commitments from Microsoft. You should verify details with the latest Microsoft documentation or product teams before making any architectural or investment decisions.

MCP server inside Copilot Studio

Microsoft integrates **Microsoft Cloud PC** server capabilities directly within Copilot Studio. This opens up the potential for creating more sophisticated, automated security workflows that blend logic, reasoning, and AI-powered actions without leaving the low-code/no-code development environment.

With this development, security teams can design custom playbooks that trigger based on alerts, incidents, or compliance gaps. These workflows can now be embedded directly into broader Microsoft 365 and Defender environments. And eventually, this could allow MSSPs to manage automated response logic at scale across multiple tenants.

Security Copilot for MSSPs: current limitations and future potential

Currently, Security Copilot is only supported for Sentinel customers who manage tenants via Azure Lighthouse. This is a limiting factor for MSSPs who work with other Microsoft security tools across multiple clients, such as Defender for Endpoint, Identity, Cloud Apps, and so on.

Microsoft is expected to expand Copilot's reach to Defender XDR workloads and cross-tenant use. In the future, multi-tenant support will be essential for MSSPs to scale intelligent detection and response. We'll likely see new API hooks and management layers for Copilot to function as a true MSSP tool as well.

Access, capability, and model improvements

Microsoft continues to iterate on Security Copilot's core AI models, aiming to gradually improve both capability and accuracy. We can expect developments and changes in various areas. For instance, access may become agentic, allowing non-human permission models per use case or team. Model quality will likely improve with more domain-specific training, such as security incident handling, and industry-specific patterns. We can also expect future Copilot tiers. Basic query support today could be developed into full remediation agents (once we trust them). A range of agents by Microsoft is just the start, completing basic tasks, such as conditional access analysis.

Protecting your IP as an MSP

Like many MSPs, you will be rightly concerned about **intellectual property** (**IP**), especially when designing detections, automation rules, or proprietary threat models. To protect your IP as an MSP, Microsoft is likely to offer tenant-level isolation of logic and configuration, so MSP playbooks aren't exposed to customers or other partners. Microsoft could also create secure export/import capabilities for logic apps, detections, and Copilot agents and store customer-specific logic and telemetry separately, with clear ownership models. This will be especially important as Copilot agents begin automating actions. We will want attribution and control.

Secure exchange: agents and the security marketplace

A new secure marketplace is expected to emerge as part of Microsoft's broader AI strategy. Think of this as the "App Store" for security agents, where verified tools, Copilot plugins, and automation agents can be published and shared.

You can expect the following key features:

- Signed and verified agents (like Visual Studio extensions today)
- Clear ownership and metadata attached to published security logic
- Possible revenue-sharing models for MSPs and developers who publish tools

This will create a secure exchange between Microsoft, partners, and customers, minimizing risk while enabling innovation.

Security agents via Visual Studio dev packages

Microsoft is working toward enabling developers to build Security Copilot agents using familiar tools such as Visual Studio and the Azure Developer CLI. This includes having a formal Agent SDK, supporting languages such as C#, Python, and **Kusto Query Language** (**KQL**); Visual Studio templates to create, test, and deploy security logic and Copilot responses; and marketplace integration to package and publish agents. This will give security developers and DevSecOps teams a real platform to contribute logic, threat hunting queries, and smart automation across environments.

Table 10.1 summarizes the changes and their potential impact:

Topic	What's Changing	The Impact
Copilot Studio + MCP	Custom security workflows	Automate incident response with low-code
Security Copilot for MSSPs	Only Sentinel + Lighthouse supported	Needs full multi-tenant, multi-product support
Model Access & Capability	Gradual AI model improvements	More accurate, useful security insights
MSP IP Protection	Expected tenant-level isolation	Keep your logic private and portable
Secure Agent Marketplace	AI-powered agent store	Share, monetize, and verify security tools
Visual Studio Agent SDK	Dev tools for Copilot agents	Build and test advanced security automations

Table 10.1: Anticipated Security Copilot releases

While these are quite futuristic, let's look at what's in store next for Microsoft Sentinel and Defender XDR.

Future outlook: Microsoft Sentinel and Defender XDR

Microsoft is undergoing a major transformation in how its security tools operate and interact, especially Sentinel and Defender XDR. These upcoming changes will impact not just technical capabilities, but also how we talk about cost, licensing, and architectural decisions with customers. With the brief insight we have, we have compiled the following sections on the possible and potential changes we can anticipate and what it means practically for organizations and security teams.

A more unified console: Sentinel inside Defender XDR

By June 2026, Microsoft Sentinel will be fully accessible through the Defender XDR console. This marks a clear step toward Microsoft's broader vision of a unified security operations platform.

For customers, it would mean the following changes:

- Seamless integration between SIEM (Sentinel) and XDR (Defender) tools
- Faster and more coordinated incident investigation and response
- Simpler, more intuitive workflows and fewer portals to manage



Sales insight

This change enables a much stronger pitch for *single-pane-of-glass* security operations. But positioning the value isn't enough. Expert configuration will be key to realizing that benefit.

Entering the Fabric-based security data lake

Microsoft is also introducing Fabric-based data lakes, which are a powerful new platform to store and analyze vast volumes of telemetry across endpoints, identities, cloud resources, applications, and network data.

This is foundational for tools such as Security Exposure Management and Microsoft Security Copilot, which require large, rich, and context-aware datasets to function optimally. Think of this as the shift to AI-native security, where automation, real-time context, and smart analytics drive responses with minimal manual effort. The data lake will power things such as risk models, Copilot agents, and advanced hunting. It is also important to note that the Fabric data lake adds depth, not speed.

Changing where and how data is stored and processed

The changes in data storage and processing **do not replace Sentinel**, but they shift the overall architecture. The following table compares the new and old data approaches of Microsoft:

Item	Legacy Approach	New (Fabric-Based) Approach
Data Ingestion	Direct to Sentinel; billed per GB/day	Ingested into Fabric-based data lake; more flexible cost

Item	Legacy Approach	New (Fabric-Based) Approach
Analytics	Always-on KQL rules on "hot" data	Batch or hourly Copilot agents using lake data
Real-Time Needs	Sentinel required (NRT, scheduled rules, real-time response)	Fabric lake not real-time; supports Co-pilot and modeling

Table 10.2: New versus old data approach

The key takeaway is that the role and importance of Sentinel remain critical for real-time alerts and detections, immediate incident response, and scheduled analytics on hot data (up to 90 days).

Let us understand how this impacts costs. We expect new pricing models to make data lake ingestion more affordable than traditional Sentinel-only models.

The following things may change:

- Lower ingestion costs to drive Fabric adoption
- Eventual ability to query Fabric data directly from Sentinel
- New SKUs or pricing tiers focused on Copilot and data lake usage

How can customers optimize costs? Many organizations aren't yet taking full advantage of the savings already available. Underused strategies such as tiered discounts starting at 100 GB/day, offering better per-GB rates at scale, and Microsoft Sentinel Commitment Plans (P3), which offer 12-month capacity commitments at lower prices, can help you optimize these costs.



Only commit to these plans once your log sources are known and stable.

What sales and security teams should be saying

Expecting these changes in Microsoft's security platform, sales and security teams should be able to clearly articulate what's changing and what isn't. This ensures that the right expectations, architecture, and investment decisions are set, which will contribute to better security and stronger customer relationships. The following points will help you start clear and helpful discussions with customers:

- *Sentinel is now part of Defender XDR*: Sales and security teams should position Microsoft Sentinel as a core capability within Microsoft Defender XDR, and emphasize tighter integration while retaining full SIEM functionality and expanding supported use cases.

- *The data lake enhances, doesn't replace, Sentinel*: Clarify that the Microsoft Security data lake enhances Sentinel rather than replacing it. It is great for AI-driven analysis, summarized rules, and proactive analytics.
- *Real-time still needs Sentinel*: Ensure that real-time detection and response still rely on Sentinel, as batch-based analytics alone cannot identify or stop live threats.
- *Costs are shifting*: Cost conversations should reflect that data ingested into Fabric-based services will be cheaper and more efficient to analyze.
- *You still need expert SIEM engineering*: Reinforce that expert SIEM engineering remains essential, as thoughtful filtering, rule design, and architecture decisions are key to controlling ingestion volumes and overall cost while maximizing security outcomes.



Introducing the Security Copilot book by Rod Trent and Bi Yue Xu

This chapter is part of a broader industry conversation inspired by Rod Trent's pioneering work on Microsoft Security Copilot. His Security Copilot book brings together insights from real-world SOC professionals, engineers, and security leaders who are reshaping threat response through AI-driven tools.

Where his Security Copilot book offers deep dives into tactical workflows, real-world use cases, and Copilot prompt engineering, this chapter complements that journey by helping you establish the right foundation, ensuring your Copilot environment is not only deployed correctly but also secure, scalable, and integrated from day one.

Summary

This chapter introduced you to Microsoft Security Copilot as an AI-powered assistant designed to help security teams operate at machine speed. We discussed the foundational requirements for deployment, including Azure subscriptions and **Security Compute Units (SCUs)**. You learned how to provision SCUs, configure access roles using Entra ID and Azure RBAC, and manage plugin-level permissions using the least privilege model. The chapter also highlighted how Copilot enables natural language prompts, interactive investigations, reusable promptbooks, and secure session sharing across teams and tenants. For MSSPs, it examined current limitations and previewed the future of multi-tenant Copilot support, agent-based automation, and IP-protected workflows. With this chapter, we reinforced how Security Copilot extends the capabilities of Defender and Sentinel by looking toward the developing field of AI-augmented security operations.

Get This Book's PDF Version and Exclusive Extras

Scan the QR code (or go to `packtpub.com/unlock`). Search for this book by name, confirm the edition, and then follow the steps on the page.

***Note**: Keep your invoice handy. Purchases made directly from Packt don't require an invoice.*

UNLOCK NOW



11

Unlock Your Exclusive Benefits

Your copy of this book includes the following exclusive benefits:

- Next-gen Packt Reader
- DRM-free PDF/ePub downloads

Follow the guide below to unlock them. The process takes only a few minutes and needs to be completed once.

Unlock this Book's Free Benefits in 3 Easy Steps

Step 1

Keep your purchase invoice ready for *Step 3*. If you have a physical copy, scan it using your phone and save it as a PDF, JPG, or PNG.

For more help on finding your invoice, visit `https://www.packtpub.com/unlock-benefits/help`.



Note: If you bought this book directly from Packt, no invoice is required. After *Step 2*, you can access your exclusive content right away.

Step 2

Scan the QR code or go to `packtpub.com/unlock`.

On the page that opens (similar to *Figure 11.1* on desktop), search for this book by name and select the correct edition.

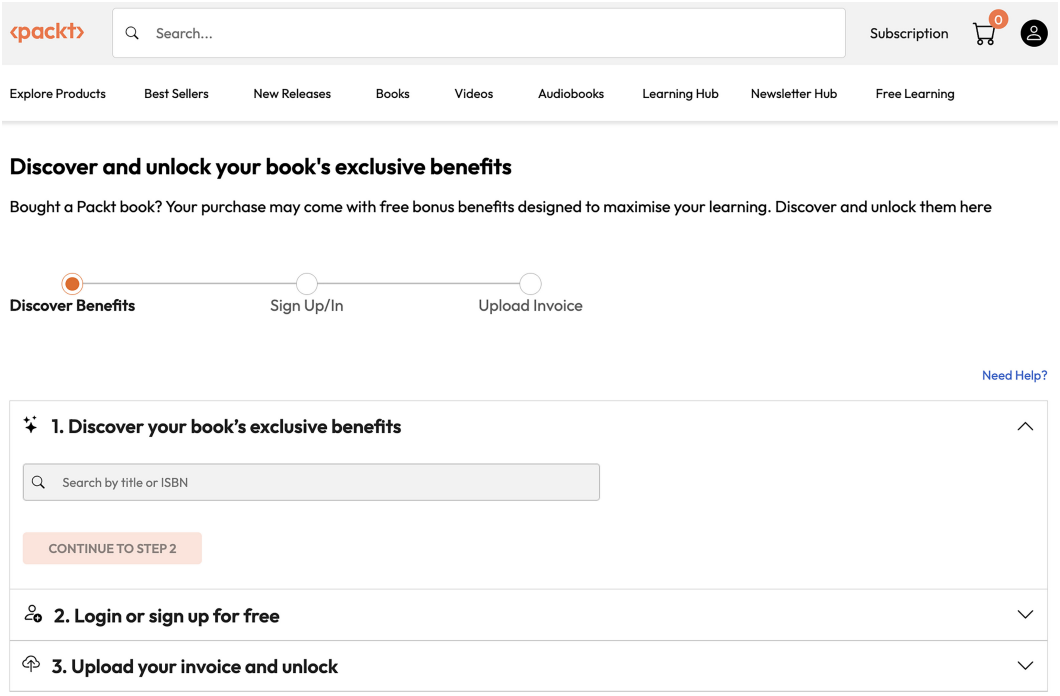


Figure 11.1: Packt unlock landing page on desktop

Step 3

After selecting your book, sign in to your Packt account or create one for free. Then upload your invoice (PDF, PNG, or JPG, up to 10 MB). Follow the on-screen instructions to finish the process.

Need help?

If you get stuck and need help, visit `https://www.packtpub.com/unlock-benefits/help` for a detailed FAQ on how to find your invoices and more. This QR code will take you to the help page.





Note: If you are still facing issues, reach out to `customercare@packt.com`.



packt.com

Subscribe to our online digital library for full access to over 7,000 books and videos, as well as industry leading tools to help you plan your personal development and advance your career. For more information, please visit our website.

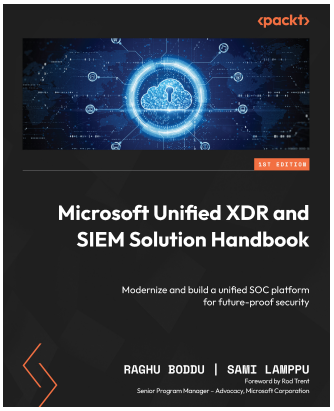
Why subscribe?

- Spend less time learning and more time coding with practical eBooks and Videos from over 4,000 industry professionals
- Improve your learning with Skill Plans built especially for you
- Get a free eBook or video every month
- Fully searchable for easy access to vital information
- Copy and paste, print, and bookmark content

At www.packt.com, you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on Packt books and eBooks.

Other Books You May Enjoy

If you enjoyed this book, you may be interested in these other books by Packt:



Microsoft Unified XDR and SIEM Solution Handbook

Raghu Boddu, None Lamppu

ISBN: 978-1-83508-685-8

- Optimize your security posture by mastering Microsoft's robust and unified solution
- Understand the synergy between Microsoft Defender's integrated tools and Sentinel SIEM and SOAR
- Explore practical use cases and case studies to improve your security posture
- See how Microsoft's XDR and SIEM proactively disrupt attacks, with examples
- Implement XDR and SIEM, incorporating assessments and best practices
- Discover the benefits of managed XDR and SOC services for enhanced protection

Packt is searching for authors like you

If you're interested in becoming an author for Packt, please visit `authors.packtpub.com` and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Share your thoughts

Now you've finished *Unified SecOps Playbook*, we'd love to hear your thoughts! If you purchased the book from Amazon, please click here to go straight to the Amazon review page for this book and share your feedback or leave a review on the site that you purchased it from.

Your review is important to us and the tech community and will help us make sure we're delivering excellent quality content.

Index

A

Active Directory Domain Services (AD DS) 109

Admin-on-Behalf-Of (AOBO) 103

Advance Data Residency
- reference link 316

Advanced Persistent Threats (APTs) 3

AI, in cybersecurity
- future 38

AI integration 464

All-in-One v2 deployment tool 414, 415
- customization and production readiness 415

American Institute of Certified Public Accountants (AICPA) 54

Anomalies 327, 328

anomaly-based rules 353

Artificial Intelligence (AI) 1

assessments
- continuous vulnerability management program 25, 26
- current security posture, assessing 17
- DevOps approach 31
- DevSecOps approach 31
- security assessment and gap analysis 17
- strategic roadmap, for security enhancements 28
- vulnerability scanning 23

authentication and delegated access model 449
- plugin permissions 450, 451
- Security Copilot role-based access 450

automated security tools 34

Automation 370, 371
- Logic Apps 371

Azure Arc 30

Azure Blueprints 33

Azure consumed revenue (ACR) 84

Azure DevOps 33
- Maester repository, setting up 174-177
- used, for automating tenant monitoring 173, 174

Azure DevOps pipeline
- executing 437-439

Azure DevOps project
- creating 416
- custom RBAC role for deployments, assigning 427, 428
- custom RBAC role for deployments, creating 427, 428
- Entra ID Security Admin service connection, configuring 423

pipeline, running 432, 433
pipeline, saving 432, 433
pipeline variables, creating 428-431
repository, importing 417
resource group, creating 419
role assignment, with Privileged Identity Management (PIM) 424-426
service connection, configuring 420-423
setting up 416, 440
standard role assignment 423

Azure Firewall 75

Azure fraud notifications 100

Azure Front Door 75

Azure Lighthouse 104, 129, 243
ARM template best practices 248
ARM template, deploying 249-252
ARM template, reviewing 245-248
best practices 244
cross-tenant access, establishing with 245
deployment options 252

Azure Logic Apps Sentinel templates 135

Azure Monitor 33

Azure Monitor Agent (AMA) 139

Azure Networking 30

Azure pipeline
client secret, creating 191
creating 192-197
Entra ID application registration, creating 182-188
permissions, granting to Azure role-based access control 189, 190
permissions, granting to Exchange Online 189
setting up 182

Azure Policy 33

Azure RBAC 302

Azure Resource Manager (ARM) templates 32

Azure subscription 116, 448
best practices 117
structuring 117, 118
subscription models, selecting 116

Azure Virtual Desktop 29

Azure Virtual WAN 75

B

B2B client-side onboarding process 261
access package security groups, creating 269-273
active roles, assigning 279-281
authentication context and conditional access policy, creating 264
Azure/Microsoft Sentinel role assignments 274, 275
conditional access authentication context 291, 292
external access governance setup 286-290
external (home tenant) user experience 294-298
external identity and cross-tenant access, configuring 261, 263
home tenant, adding as connected organization 266-268
initial B2B MSSP sponsors, inviting 265
process, repeating for required roles 278
role assignments, adding to groups 274
roles, making available, via authentication context 276-278
URBAC Defender roles 282-286

Bicep 413

branch-based model 398

brownfield migration 148
implementing 149

C

Center for Internet Security (CIS) 9, 55
automated security assessments 56
Azure, securing with CIS Benchmarks 56, 57
collaboration, with Microsoft 56
core components 55, 56
examples 56
centralized workspace architecture 118
design considerations and trade-offs 118, 119
selection scenario 119
CIA triad 12
applying, to security strategy 14-16
principles 13, 14
CIS Amazon Web Services (AWS) Foundations Benchmark 56
CIS Benchmarks
reference link 55
CIS Controls v8.1 10
CIS Microsoft 365 Foundations Benchmark
reference link 57
CIS Microsoft Azure Foundations Benchmark 56
reference link 57
Claiming Partner of Record (CPOR) 79, 101
claiming 102
client blob container
creating, in storage account 443
client-specific pipelines and solution-driven 399
audit-ready 401
centralized reusability 401
configuration file (Config.yaml) 401, 402
deployment pipeline file (sentinel-deploy-<id>.yml) 402-404
per-tenant CI/CD pipelines 401
Cloud Adoption Framework (CAF) 57, 60
key components 60, 62
versus Well-Architected Framework (WAF) 64
Cloud Center of Excellence (CCoE) 62
Cloud Security Posture Management (CSPM) 64
cloud solution provider (CSP) 97
Coalition for Content Provenance and Authenticity (C2PA) 38
Co-Management 409
Common Event Format (CEF) 139
comprehensive security assessment
conducting 17, 18
content
deploying, via Workspace Manager 408, 409
content hub solutions
customizing 341-346
Defender XDR solution, accessing 336-341
deploying 335, 336
optimizing 341-346
contextual authentication 67
continuous vulnerability management program
automated patch management, for operating systems and applications 26
effective remediation process, creating 26
establishing 25, 26
exposed assets, monitoring externally 28
regular vulnerability scans, conducting 27
vulnerabilities, remediating promptly 28
critical security controls
for effective threat mitigation 9
cross-tenant access
establishing, via Azure Lighthouse 245
customer-managed keys (CMKs) 137

Cyber Assessment Framework (CAF) 44

Cyber Essentials 47

Cyber Essentials Plus 48

cybersecurity 50
domains 51

Cybersecurity and Infrastructure Security Agency (CISA) 164

cybersecurity, best practices 8
adapting, to organization's unique needs 10
comprehensive cybersecurity framework, implementing 8, 9
critical security controls 9, 10

cybersecurity frameworks 42
aligning 57
Center for Internet Security (CIS) 55
Cyber Essentials 47
Cyber Essentials Plus 47
General Data Protection Regulation (GDPR) 52
ISO/IEC 27001 49
National Institute of Standards and Technology (NIST) 42
NCSC CAF 44
System and Organizations Controls Type 2 (SOC 2) 54, 55

Cybersecurity Investment (CSI)
versus Microsoft Commerce Incentives (MCI) 93, 94

Cybersecurity Investment (CSI) Program 92
benefits 93
business impact 93

D

data, applications, assets, and services (DAAS) 72

data assets
protecting 30

data collection rule (DCR) 139, 278

data connector planning, in Sentinel 138
aligning, with compliance and retention requirements 140
high-value data sources, prioritizing 138, 139
SOC optimization tool, using 139

data encryption 68

Data Loss Prevention (DLP) 74

data retention 154, 155
budget protection 156
controls 156
cost optimization 156
coverage optimization 156
filtering 156
performance optimization 156

data tiers, Sentinel 159
Analytics tier 159
Sentinel data lake 159
shift from older models 160

decentralized workspace architecture 119, 120
design considerations and trade-offs 120
selection considerations 120, 121

Defender for Cloud 33

Defender for Cloud Apps 30

Defender XDR 143, 467
capabilities 143, 144

Defender XDR Unified platform 282

Delegated Admin Privileges (DAP) 98, 253

deployment roadmaps, Sentinel 147
brownfield migration 148, 149
greenfield deployment 147
MSMMP 152
phase-based migration framework 150

side-by-side strategy 149
strategic considerations 150
deployment via Terraform 435
Azure DevOps pipeline integration 435
state management strategy 436
testing and validation 436
detection use case matrix 156
DevOps 31
DevSecOps 31
key components 32
Microsoft tools and services, for implementation 33
practices, integrating 31
principles 32
DevSecOps in organization 34
building and testing 36
code commit 35, 36
operations 36
planning and development 34, 35
production 36
diagnostic logs
Anomalies 327, 328
UEBA 325, 327
Direct-Link 409
directory-based model 398
Domain-based Message Authentication, Reporting, and Conformance (DMARC) 112
DomainKeys Identified Mail (DKIM) 111
dynamic security testing 36

E

Endpoint Detection and Response (EDR) 4, 65
endpoint security solutions 68
Enterprise Agreement (EA) 116
entity mapping 353
automated response within Analytics Rules 359
automation rules 359-363
incident handling, optimizing in Microsoft Sentinel 356-358
use case 354-356
Entra ID B2B access model 259
implementing 259, 260
mandatory requirements 261
Entra ID Governance 29
Entra Permissions Management 30
Entra Private Access 29
Entra Public Access 29
European Commission's GDPR Portal
reference link 54
European Union (EU) 52
Exchange Online Protection (EOP) 110

F

Fabric-based data lake 467
foundational access controls
creating 215, 216
team-based role design for Sentinel operations 218
URBAC concept 216, 217
foundational RBAC artifacts
access package, creating 221-227
deploying 219
PIM, enabling 229-234
security groups, creating 219, 221
foundational RBAC artifacts, access package
role-based access groups, defining 228
fusion analytics rule 353

G

gap analysis 20
risk assessment methodology 21
security risks and vulnerabilities, identifying 21
security risks and vulnerabilities, prioritizing 21
General Data Protection Regulation (GDPR) 52
cybersecurity impact 52, 53
GitHub Advanced Security (GHAS) 30, 33
governance frameworks 68
granular delegated admin privileges (GDAP) 98, 100, 208
granular delegated admin privileges (GDAP) connection 253
admin relationship, creating 253-255
client environment, accessing via 259
request, finalizing 256
request, sending 256
roles, mapping to Azure AD security groups 257, 258
greenfield deployment 147
groups
organizing, in Workspace Manager 410, 411
publishing content 411, 412

H

high signal security data 159
hybrid workspace architecture 121
challenges 122, 123
design considerations and trade-offs 122
selection considerations 123
strategic planning 122, 123

I

IAC Sentinel deployment 413
deployment via Terraform 435
terraform automation and deployment architecture, for multi-tenant environments 436, 437
via Bicep 416
via code 413
IAC Sentinel deployment, via Bicep
Azure DevOps project, setting up 416
pipeline workflow 434
PowerShell script 434, 435
IAC Sentinel deployment, via code
All-in-One v2 deployment tool 414, 415
Identity and Access Management (IAM) 67, 73, 108, 113
data security and encryption 74
endpoint security 73
network security 74
security policies and governance 74
Incidents 375-377
independent software vendors (ISVs) 80
indicators of compromise (IOCs) 383
Information Security Management System (ISMS) 49, 57
infrastructure security
for hybrid environments 30
intellectual property (IP) 465
Internet Message Access Protocol (IMAP) 113
ISO 27005 22
ISO/IEC 27001 49
2022 version update 50
controls 49

J

Just-Enough Access (JEA) 8

Just-In-Time (JIT) 8, 64, 114

K

KQL-based hunting queries 381

Kusto Query Language (KQL) 152, 346, 347, 466
- detection rules, types 352
- entity mapping 353
- rule logic 351, 352

Kusto Query Language (KQL) functions 348
- advanced filters and expressions 349
- aggregations 350
- core operators and use cases 348, 349
- data enriching 350, 351
- data grouping 350
- data joining 350, 351
- data transformation 349, 350
- visualization 351

L

large language models (LLMs) 448

LAWS components
- diagnostic logs, enabling 322-324
- Microsoft Sentinel, enabling 317
- Microsoft Sentinel setup, settings 318-322
- setting up 313

Log Analytics Contributor 127

Log Analytics Workspace (LAWS) 155, 313
- creating 314-316

logical operators 349

Logic Apps 371
- use case 372-375

long-term storage 154

M

Machine Learning (ML) algorithms 2

Maester tool
- client's location, adding 177-181
- used, for automating tenant monitoring 173, 174

Managed Extended Detection and Response (MXDR) 88

managed security service providers (MSSPs) 91

managed service providers (MSPs) 80

Mean Time to Detect (MTTD) 6

Mean Time to Respond (MTTR) 6

micro-segmentation 68

Microsoft 365
- Multitenant Organizations (MTOs) 115

Microsoft 365 tenant 108
- attributes 108, 109
- configuring, for enhanced security 109
- email security, with authentication protocols 111, 112
- fundamentals 110, 111
- identity and access management (IAM) 113
- protection policies, using in EOP and MDO 112

Microsoft AI Cloud Partner Program 80
- benefits, for security providers 80, 81
- member 81, 82
- Solutions Partner 82, 83

Microsoft-certified MXDR partner program 88
- advantages 88
- assessment and verification process 89
- requirements 89
- technical requirements for verification 90

Microsoft Cloud PC 464

Microsoft Commerce Incentives (MCI) 93
versus Cybersecurity Investment (CSI) 93, 94

Microsoft Customer Agreement (MCA) 116

Microsoft Defender Application Guard 29

Microsoft Defender for APIs 30

Microsoft Defender for Cloud 30, 33, 76

Microsoft Defender for Cloud Apps 29

Microsoft Defender for Endpoint (MDE) 3, 4

Microsoft Defender for Identity 4, 29

Microsoft Defender for Office 365 (MDO) 5, 9, 110

Microsoft Defender Vulnerability Management 23
key features 24

Microsoft Defender XDR 1-3, 29
AI, used for transforming modern security landscape 3, 4
future outlook 7
Security Copilot initiation 6
unified security approach and ecosystem 4, 5

Microsoft Defender XDR permissions 303
URBAC role for Defender XDR investigations 303

Microsoft Defender XDR workload
consideration 303, 304

Microsoft Digital Defense Report 2024 12

Microsoft Endpoint Configuration Manager 24

Microsoft Entra admin center
reference link 114

Microsoft Entra Conditional Access 29

Microsoft Entra ID 29
security defaults 113

Microsoft Entra ID Protection 29

Microsoft FastTrack 96
partners 96

Microsoft, in cybersecurity
future 38

Microsoft Intelligent Security Association (MISA) 91
membership 92
reference link 92

Microsoft Intune 29

Microsoft MTOs 143

Microsoft Partner Network (MPN) 102

Microsoft Partners 253

Microsoft's Cybersecurity Reference Architecture (MCRA) 8

Microsoft Secure Score 24
features 24

Microsoft Security advanced specializations 86
requirements 87

Microsoft Security Copilot 6, 7
advantage 37

Microsoft Sentinel 1, 5, 29, 346, 466
backend secrets, adding 441, 442
data storage and processing 467
deploying, with Terraform 440, 441
security platform 468
service connection, creating in Azure DevOps 442
storage account, creating 442
sub-tenant requirements setting up 443

Microsoft Sentinel artifacts
content, deploying via repositories 392
content deployment via repositories, setting up 392, 393

deploying 392
managing 392

Microsoft Sentinel artifacts, content deployment
deployment status, tracking 406-408
pipeline file location, making 405, 406

Microsoft Sentinel artifacts, content deployment via repositories
client-specific pipelines 398-400
connecting, via client tenant 394, 395
directory-based model, versus branch-based model 398
repository connection, setting up 393, 394
repository details, adding 396, 397
repository setup 397
solution-driven deployment 398-400

Microsoft Sentinel Attack Range 388

Microsoft Sentinel Contributor 127

Microsoft Sentinel deployment plan
executing 313
LAWS components, setting up 313
security setup 306

Microsoft Sentinel deployment plan, security stages
collection and enhance visibility, optimizing 308
data collection 308, 309
data collection, innovating 310
foundation, building 306, 307
SOC analysts with proactive tools 312
threat intelligence detection 311

Microsoft Sentinel foundations
Azure Lighthouse scenarios 214, 215
B2B scenarios 214, 215
designing, with Azure subscriptions and management groups 208
log collection impact 209
management groups 208
subscription, selecting 209, 210
unified XDR access 213
URBAC 212

Microsoft Sentinel foundations, subscription
management group, creating 210, 211
new subscription, creating 212

Microsoft Sentinel GitHub repository 134

Microsoft Sentinel Migration
key components 94, 95

Microsoft Sentinel Migration and Modernization Program (MSMMP) 152
best practices 152
tools and resources 153

Microsoft Sentinel Playbook Operator 302

Microsoft Sentinel pricing model 158
archive and long-term retention 160
costs 158
data tiers 159

Microsoft Sentinel Reader 127, 302

Microsoft Sentinel Responder 127, 302

Microsoft Sentinel SIEM deployment plan 306

Microsoft Sentinel SIEM permissions 302

Microsoft Sentinel solutions
content hub solutions, deploying 335, 336

Microsoft Sentinel Zero Trust (TIC 3.0) 77

Microsoft tenant
assessing, with ScubaGear 164

Microsoft Tenant Organizations (MTO) 214

Migration and Modernization Program (MMP) 94

MITRE ATT&CK framework 386, 388

monthly active users (MAU) 84

Multi-Factor Authentication (MFA) 12, 67, 109

multiple Sentinel workspaces
managing 304, 305
multitenant management, with MTO and Defender XDR
best practices 145
considerations 143, 144
limitations 144
Multitenant Organizations (MTOs) 115

N

National Cybersecurity Centre (NCSC) 22, 44
reference link 54
National Institute of Standards and Technology (NIST) 42
NCSC CAF 44
adoption and implementation 46
benefits 46
key objectives 45
reference link 46
near-real-time (NRT) 352
NIST CSF
core functions 42-44
NIST CSF 2.0 44
reference link 44
resources 44
NIST Cybersecurity Framework (CSF) 12
NIST SP 800-30 21
N-Tier 409

O

OCTAVE
using 22
ongoing threat detection monitoring 16
Online Services Advisor (OSA) 102
operational technology (OT) environments 120
operational technology (OT) systems 3
OS Credential Dumping (T1003) 387
overage capacity 449

P

Partner Admin Link (PAL) 79, 103
Azure Lighthouse deployment 104
setting up 103, 104
Pay-as-you-go (PAYG) 116, 160
Personally Identifiable Information (PII) 15
phase-based migration framework 150
design 151
discovery 150
implementation 151
operationalization 151
Post Office Protocol (POP) 113
practical automation
use cases 134
Privileged Access Management (PAM) 18
Privileged Identity Management (PIM) 101, 114, 207, 305
Azure roles, assigning via IAM 237
enabling 229-234
roles 235
roles, assigning in unified security operations platform 235, 237
Promptbooks 448
provisioned capacity 449

Q

quantum computing 76
quantum key distribution (QKD) 76
quantum-resistant cryptographic algorithms 76

R

RBAC, for Microsoft Sentinel
access control 126
access, managing with Azure Lighthouse 128, 129
audit 130
benefits 126
custom roles usage 129
designing 126
life cycle, building 128
review 130
role assignment pitfalls, avoiding 128
rotate 130
segment access, by scope 129, 130
tiered SOC model 127

RBAC framework
common role-based requirements 240
home tenant role matrix, example 239, 240
integration method, selecting 241
multitenant role assignment 241
setting up 239

regular security review cycle 16

risk assessment methodology 21, 22

role-based access control (RBAC) 71, 448

S

ScubaGear
license requirements 164
output 172, 173
PowerShell, setting up 167-169
running 164, 169-171
script, downloading 164-167
used, for assessing Microsoft tenant 164

SecOps platform
connecting 328-333
Microsoft Sentinel solutions, deploying 334
multi-tenant management 333, 334
security data collection 334

Secure Cloud Business Applications (SCuBA) 164

Secure DevOps Kit for Azure (AzSK) 33

Secure Future Initiative (SFI) 58
core principles 59
strategic pillars 59

Security Adoption Framework (SAF) 10
key stages 10, 11

security assessment 17

Security Compute Units (SCUs) 449
overage capacity 449
provisioned capacity 449

Security Copilot 1, 464
Azure portal, using 456-458
capacity, provisioning 452-455
default environment, configuring 458-462
developments and changes 465
for MSSPs 464
intellectual property (IP) 465
MCP server 464
provisioned capacity, deleting 462, 463
running 451
security agents, via Visual Studio dev packages 466
security marketplace 465

security defaults, in Microsoft Entra ID 113
IAM strategy, expanding 114, 115
managing 114

security-focused partner programs
exploring 88
Microsoft-certified MXDR partner program 88
Microsoft Intelligent Security Association (MISA) 91

Security Information and Event Management (SIEM) platform 5

Security Operations Centers (SOCs) 65

Security Operations (SecOps) platform 1

Security orchestration, Automation, and Response (SOAR) 18

security partner 97
- Azure fraud notifications 100
- customer access models 98
- requirements 97

Sender Policy Framework (SPF) 111

Sentinel pre-deployment checklist
- automation strategy, defining 133, 134
- cluster guidance, for high-volume ingestion 136, 137
- data connector planning 138
- Defender and Microsoft 365 integration 132, 133
- deployment team, assembling 132
- governance, documenting 135
- ingestion and budget, estimating 131, 132
- maintenance 135
- roles 135
- strategic automation 140
- use cases, identifying 131
- use cases, prioritizing 131
- workspace architecture, selecting 132
- workspace naming conventions 136

Set SentinelContent.ps1 434

shift-left security 31, 32

Simple Mail Transfer Protocol Authentication (SMTP AUTH)) 113

Single Sign-On (SSO) 67

small and medium businesses (SMBs) 83
- customer success 85, 86
- designation framework 84
- MSSP or multitenant model, aligning with 86
- performance 85
- skills 85

SOC maturity 133

Software Development Life Cycle (SDLC) 31

Solutions Partner, Microsoft AI Cloud Partner Program
- Microsoft Security advanced specializations 86, 87
- security 83, 84
- small and medium businesses (SMBs) 84
- strategy value 86

standalone RBAC 305
- Privileged Identity Management (PIM) 305

Static Application Security Testing (SAST) 35

strategic automation, in Sentinel 140
- automation rules 141
- health monitoring 142
- playbooks 141
- use cases 142

strategic roadmap development, for security enhancements 28
- applications, securing against threats 30
- data assets, protecting 30
- identity management 29
- infrastructure security, for hybrid environments 30
- modern SecOps, for swift response 29
- user endpoints, securing 29
- Zero Trust architecture 29

stringent access controls 68

string operators 349

sub-tenant requirements
- client blob container, creating 443
- client variables file, preparing 443, 444

pipeline, executing 444, 445
setting up 443

summary rule 364
creating 365, 366
tips, using 366
watchlists, deploying 367-370

System and Organizations Controls Type 2 (SOC 2) 54, 55
implementing 54

T

tenant monitoring
automating, with Azure DevOps 173, 174
automating, with Maester tool 173, 174

Terraform 413

terraform automation and deployment architecture
for multi-tenant Sentinel environments 436, 437

threat hunting 380, 381
advanced hunting, in Defender XDR 382, 383

threat intelligence 383-386

threat intelligence platforms (TIPs) 385

Threat Intelligence teams 37

threat modeling 34

tools, for ingestion strategy
cost estimation workbooks 158
free trials and credits 158
Microsoft Sentinel content hub 158
Sentinel Optimization Workbook 158

Trusted Internet Connections (TIC) 3.0 model 77

U

UK Information Commissioner's Office (ICO)
reference link 54

Unified Microsoft Defender Portal
permissions 302

Unified Role-Based Access Control (URBAC) 207, 212

Uniform Resource Names (URNs) 189

URBAC Defender roles 282

User and Entity Behavior Analytics (UEBA) 325, 327

V

vulnerability assessment
with Microsoft Defender Vulnerability Management 24

vulnerability management 22
vulnerabilities, prioritizing with Microsoft Secure Score 24

vulnerability scanning
features 23

W

Well-Architected Framework (WAF) 57, 62, 63
pillars 63
versus Cloud Adoption Framework (CAF) 64

Wireless Access Points (WAPs) 18

Workbooks 378-380

workspace architecture, Microsoft Sentinel
access control 124
centralized workspace architecture 118
cost 124
decentralized workspace architecture 119, 120
hybrid workspace architecture 121
scalability 125

Workspace Manager 145, 408
for MSSPs 145-147
groups, organizing 410, 411
used, for deploying content 408-410

Z

Zero-hour Auto Purge (ZAP) 380
Zero Trust 8, 64
architecture 29
defining 64
fundamental principles 8
future 76, 77
implementing 64
modern cybersecurity model 64
reactive to proactive 65
with Microsoft solutions 75
Zero Trust architecture 67
awareness and training 69
data security 68
device security 68
identity verification 67
implementing 67
network security 68
security policies and governance 68
Zero Trust Assessment Tool 197
first-time module, installing 199-203
Maester test results 204
prerequisites 197, 199
telemetry setting 203
Zero Trust challenges 69
cost and complexity 70
cultural shift 69
legacy systems 69
regulatory compliance 70
security measures, integrating 70
Zero Trust Network Access (ZTNA) 74
Zero Trust principles 66
assume breach 66
explicit verification 67
least privilege access 66
micro-segmentation 67
verification 66
Zero Trust strategy
environment 71, 72
planning 70
Zero Trust technologies and solutions
exploring 73
Identity and Access Management (IAM) 73