MICROSOFT CERTIFICATION — COMPLETE STUDY GUIDE

MD-102

Microsoft Endpoint Administrator

A Practical, Interview-Focused Enterprise Study Guide

Section 1 — Deploy Windows Client

Section 2 — Manage Identity and Compliance

Section 3 — Manage, Maintain, and Protect Devices

Section 4 — Manage Applications

Final Reference Appendix

Real-World Administration • Troubleshooting • Hands-On Labs • Interview Preparation • Cheat Sheets • 50 Interview Questions

Master Table of Contents

Section 1 — Deploy Windows Client

Windows deployment methods · Autopilot · provisioning packages · subscription activation · Entra Join · Hybrid Join · Windows Update · cloud-based deployment

Includes consolidated interview preparation and a symptom-first troubleshooting reference.

Section 2 — Manage Identity and Compliance

Microsoft Entra ID · device registration · compliance policies · Conditional Access · authentication methods · passwordless · BitLocker · compliance reporting

Includes consolidated interview preparation and a symptom-first troubleshooting reference.

Section 3 — Manage, Maintain, and Protect Devices

Intune overview · enrollment · configuration profiles · administrative templates · device restrictions · endpoint security · Defender for Endpoint · remote actions · monitoring · Windows Update for Business

Includes consolidated interview preparation and a symptom-first troubleshooting reference.

Section 4 — Manage Applications

App deployment · Microsoft Store apps · Win32 apps · Microsoft 365 Apps · app protection policies · app configuration policies · application updates · application troubleshooting

Includes consolidated interview preparation and a symptom-first troubleshooting reference.

Final Reference Appendix

Exam strategy · practical lab checklist · daily tasks · Intune & Entra portal cheat sheets · common exam mistakes · last-minute revision · mock interviews · enterprise best practices · administrator checklist · PowerShell cheat sheet · 50 most asked interview questions

About this guide: This master volume combines all four MD-102 curriculum domains and a complete reference appendix into a single continuous book. Each section follows the same fixed structure — concepts, architecture, exact portal navigation, step-by-step configuration, real-world scenarios, enterprise examples, troubleshooting, best practices, security considerations, PowerShell, interview Q&A, exam notes, hands-on labs, summaries, and mini quizzes. Page numbers run continuously throughout. Each section also retains its own cover and local table of contents for easy navigation.

MICROSOFT CERTIFICATION

MD-102

Microsoft Endpoint Administrator

Section 1 — Deploy Windows Client

A Practical, Interview-Focused Enterprise Study Guide

Windows Autopilot · Entra Join · Hybrid Join · Provisioning · Subscription Activation · Windows Update

Real-World Administration • Troubleshooting • Hands-On Labs • Interview Preparation

Section 1 Contents

1. Windows Deployment Methods Overview
2. Windows Autopilot
3. Provisioning Packages
4. Subscription Activation
5. Microsoft Entra Join
6. Hybrid Microsoft Entra Join
7. Windows Update Deployment
8. Cloud-Based Deployment Strategy
9. Section 1 — Consolidated Interview Preparation
10. Section 1 — Troubleshooting Reference

How to use this section: Each topic follows a fixed structure — concepts, architecture, exact portal navigation, step-by-step configuration, real-world scenarios, enterprise examples, troubleshooting, best practices, security, PowerShell, interview Q&A, exam notes, a hands-on lab, summary, and a mini quiz. Work through the labs in a trial Microsoft 365 / Intune tenant for maximum retention.

1. Windows Deployment Methods Overview

Introduction

Before managing devices with Intune, you must get Windows onto those devices and connected to your organization's identity and management systems. **Windows deployment** is the process of installing, configuring, and enrolling Windows clients so they are ready, secure, and managed. Modern endpoint administration has shifted away from manually imaging every laptop toward **cloud-driven, zero-touch** approaches where a device configures itself the first time an employee signs in.

Why This Topic Is Important

Every company that issues Windows laptops or desktops needs a repeatable way to deploy them. The method chosen affects cost, IT workload, security, and how quickly a new employee becomes productive. In interviews and on the job you are expected to know **which method fits which scenario** — for example, why a remote-first company chooses Autopilot while a factory floor with legacy apps may still use traditional imaging.

Key Concepts

- **Traditional (thick) imaging:** A captured `.wim` image with OS, apps, and settings is pushed to devices using tools like MDT or Configuration Manager. Heavy, but full control.
- **Modern deployment / provisioning:** A clean OEM image of Windows is transformed into a corporate device using cloud configuration (Autopilot) or a provisioning package — no re-imaging.
- **Bare metal:** Installing the OS on a device with no existing operating system.
- **In-place upgrade:** Moving from one Windows version to a newer one while keeping apps and data.
- **Dynamic provisioning:** Taking the existing OEM Windows and only layering organizational identity, policies, and apps on top.
- **Zero-touch:** No IT physically handles the device — it ships from the vendor directly to the user.

Architecture / Workflow

Deployment methods sit on a spectrum from heavy IT involvement to fully automated cloud provisioning:

Method	Tooling	IT Touch	Best For
Traditional imaging	MDT, Configuration Manager (SCCM)	High — wipe and load	Standardized hardware, on-prem heavy, legacy apps
Windows Autopilot	Intune + Entra ID	Very low — zero-touch	Remote/cloud-first orgs, new OEM devices
Provisioning packages	Windows Configuration Designer	Medium — run a package	Offline sites, no network deployment, kiosks
Subscription activation	Entra ID licensing	Low — step-up edition	Upgrading Pro to Enterprise via license
In-place upgrade	Windows Update / feature update	Low	Moving Windows 10 → 11, version bumps

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Enrollment → Windows → Windows enrollment
```

```
Intune Admin Center → Devices → Windows → Windows devices
```

Real-World Scenario

Scenario: Choosing a method

A 400-person consulting firm is fully remote. New hires receive laptops shipped from a hardware vendor directly to their homes. Traditional imaging is impossible because IT never touches the device. The administrator chooses **Windows Autopilot** so the laptop configures itself over the internet when the new hire powers it on and signs in with their work account.

Enterprise Deployment Example

A multinational with 30,000 endpoints runs a **hybrid model**: corporate offices with legacy line-of-business apps still use Configuration Manager imaging and Hybrid Entra Join, while new remote and frontline workers use Autopilot with pure Entra Join. A co-management bridge lets both worlds be governed from a single console as the company migrates fully to the cloud over three years.

Best Practices

- Default to **cloud-native (Autopilot + Entra Join)** for new deployments unless a hard requirement forces hybrid.
- Standardize on a small set of hardware models to simplify drivers and testing.
- Document a deployment runbook so any technician can reproduce the process.

Security Considerations

Modern deployment improves security by enforcing identity, encryption (BitLocker), and compliance from the very first boot, rather than relying on a manually hardened image that drifts over time. Always ensure devices are encrypted and compliant *before* granting access to corporate data via Conditional Access.

PowerShell Commands

```
# Check current Windows edition and build
Get-ComputerInfo | Select-Object WindowsProductName, OsBuildNumber, WindowsVersion

# Check Entra/MDM join state of a device
dsregcmd /status
```

Interview Questions and Answers

BEGINNER **What is the difference between traditional imaging and modern provisioning?**

Traditional imaging wipes the device and loads a custom OS image. Modern provisioning keeps the OEM Windows and only adds corporate identity, policies, and apps — usually with no IT physically touching the device.

INTERMEDIATE **A company is fully remote and ships laptops to homes. Which deployment method fits best and why?**

Windows Autopilot, because it is zero-touch — the device self-configures over the internet when the user first signs in, removing the need for IT to image or handle the hardware.

ADVANCED **Why might a large enterprise keep using Configuration Manager imaging alongside Autopilot?**

Legacy line-of-business apps, complex driver requirements, on-prem-only resources, or existing investment in imaging infrastructure. Co-management lets them run both while migrating workloads to the cloud gradually.

MD-102 Exam Notes

Exam Focus

Know which deployment method maps to which business requirement. The exam favors **cloud-native** answers (Autopilot, Entra Join) unless the scenario explicitly mentions on-prem dependencies (legacy apps, on-prem resource access), which point toward Hybrid Join or Configuration Manager.

Hands-On Lab

Lab 1.1 — Survey deployment readiness

1. Sign in to the Intune Admin Center. 2. Navigate to *Devices* → *Windows* and note how many devices are present. 3. Open *Devices* → *Enrollment* → *Windows* and identify which enrollment methods are configured. 4. On a test machine, run `dsregcmd /status` and record the AzureAdJoined and DomainJoined values.

Summary

Windows deployment ranges from heavy traditional imaging to zero-touch cloud provisioning. Modern endpoint administration favors Autopilot and Entra Join, while hybrid environments retain imaging for legacy needs. The right choice depends on connectivity, app dependencies, and how much IT can physically handle devices.

Mini Quiz

1. Which deployment method requires no physical IT contact with the device?
2. Name two tools used for traditional imaging.
3. What command shows a device's Entra join status?
4. When would you still choose Hybrid Entra Join over pure cloud?
5. What is an in-place upgrade?

Answers: 1) Windows Autopilot. 2) MDT and Configuration Manager (SCCM). 3) `dsregcmd /status`. 4) When devices need on-prem AD/resource access or legacy app/GPO dependencies. 5) Upgrading Windows version while keeping apps and data.

2. Windows Autopilot

Introduction

Windows Autopilot is Microsoft's cloud-based, zero-touch deployment technology. Instead of building and pushing an OS image, Autopilot takes the Windows that already ships on a new device from the manufacturer and automatically transforms it into a fully configured, corporate-managed device the moment the user connects to the internet and signs in with their work account. The device's unique **hardware hash** is registered with your tenant, so when it boots, Microsoft's cloud recognizes it and applies your deployment profile.

Why This Topic Is Important

Autopilot is the flagship modern-deployment feature and one of the most heavily tested areas of MD-102. In the real world it eliminates imaging, slashes provisioning time from hours to minutes, and lets devices ship directly from a vendor to a remote employee. Almost every cloud-first organization uses it, so interviewers probe it deeply.

Key Concepts

- **Hardware hash (hardware ID):** A unique fingerprint of the device collected from firmware. Used to identify and register the device with Autopilot.
- **Autopilot deployment profile:** Settings that control the out-of-box experience (OOBE) — join type, naming, what screens the user sees, whether they get admin rights.
- **OOBE:** Out-of-Box Experience — the first-run setup screens after powering on a new Windows device.
- **Enrollment Status Page (ESP):** A progress screen that blocks the user from the desktop until required apps, policies, and certificates are applied.
- **Deployment modes:** User-driven, Self-deploying, Pre-provisioning (formerly White Glove), and Autopilot for existing devices.
- **Dynamic group:** An Entra group whose membership is automatically built from a rule (e.g., all Autopilot devices), used to target profiles.

Architecture / Workflow

The end-to-end Autopilot flow:

1. OEM or admin captures the device's **hardware hash** and uploads it to Intune (or the OEM registers it for you).
2. The device appears under *Windows enrollment → Devices* and joins a dynamic group.
3. An **Autopilot deployment profile** is assigned to that group.
4. User unboxes the device, connects to Wi-Fi, and the OOBE contacts Microsoft.
5. Microsoft matches the hardware hash, downloads the profile, and customizes OOBE.
6. The device performs **Entra Join** and auto-enrolls in Intune.
7. The **Enrollment Status Page** tracks apps/policies until complete.
8. The user lands on a ready-to-work desktop.

Autopilot Deployment Modes

Mode	How It Works	Typical Use
User-driven	User signs in with work account; device joins and enrolls under that user.	Standard employee laptops
Self-deploying	No user credentials needed; device provisions itself automatically (requires TPM 2.0).	Kiosks, digital signage, shared/ Teams Room devices
Pre-provisioning	IT/OEM pre-loads apps and policies (the long part) before handing a near-ready device to the user.	Faster end-user experience, reseller staging
Existing devices	Autopilot applied during a Configuration Manager OS task sequence on already-owned hardware.	Reimaging existing fleet into cloud model

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Enrollment → Windows → Windows enrollment → Deployment Profiles
```

```
Intune Admin Center → Devices → Enrollment → Windows → Windows enrollment → Devices (Autopilot devices)
```

```
Intune Admin Center → Devices → Enrollment → Windows → Windows enrollment → Enrollment Status Page
```

Step-by-Step Configuration

A. Register a device

1. On the target device, collect the hash with PowerShell (see PowerShell section) to produce a CSV.
2. Go to *Windows enrollment* → *Devices* → *Import* and upload the CSV.
3. Wait for sync (can take up to 15 minutes); the device appears in the Autopilot devices list.

B. Create a deployment profile

1. Go to *Windows enrollment* → *Deployment Profiles* → *Create profile* → *Windows PC*.
2. Name it (e.g., "AP - User Driven - Corporate").
3. Set **Deployment mode** = User-Driven; **Join to Entra ID as** = Microsoft Entra joined.
4. Configure OOBE: hide license terms, hide privacy settings, hide change account options, set user account type to Standard, apply device name template (e.g., `CORP-%SERIAL%`).
5. Assign the profile to a dynamic device group containing your Autopilot devices.

C. Configure the Enrollment Status Page

1. Go to *Enrollment Status Page* → *Create* (or edit Default).
2. Show app and profile installation progress = Yes.
3. Block device use until all apps and profiles are installed = Yes.

4. Optionally specify required apps that must install before access.

Dynamic group rule example

```
(device.devicePhysicalIds -any (_ -contains "[ZTDId]"))
```

This rule automatically captures every Autopilot-registered device.

Real-World Scenario

Scenario: Remote new hire

A new sales rep in another city is hired Monday. A laptop ships from the OEM straight to her home, already registered in Autopilot by the vendor. She powers on, joins her home Wi-Fi, signs in with her Microsoft 365 account, and the ESP installs Microsoft 365 Apps, VPN, and security policies. Twenty minutes later she has a compliant, encrypted, managed laptop — IT never touched it.

Scenario: Conference room device

A company deploys 50 Microsoft Teams Rooms. Using **Self-deploying mode**, each device provisions with no user sign-in, joins Entra, enrolls in Intune, and applies kiosk policies automatically.

Enterprise Deployment Example

A retailer with 5,000 stores standardizes on one laptop model. The OEM registers every purchased device directly into the company's tenant via the OEM's Autopilot integration, so devices never need manual hash collection. Pre-provisioning is run at the reseller's warehouse so heavy apps are pre-loaded; store staff simply sign in and are productive in two minutes. Dynamic groups route store-manager devices to one profile and floor-staff devices to a kiosk profile.

Troubleshooting

Autopilot device not receiving its profile

Symptoms: Device goes through generic OOBE, no corporate branding, asks for personal setup.

Root Cause: Device not registered, profile not assigned, or dynamic group hasn't populated yet.

Steps: Confirm the device appears under Autopilot devices; confirm "Profile status" = Assigned; verify the dynamic group membership; force a sync; ensure the device reached the internet during OOBE.

Resolution: Re-import the hash if missing, wait for group/profile assignment, then reset the device and restart OOBE.

Enrollment Status Page stuck or timing out

Symptoms: ESP hangs on "Installing apps" or shows a timeout error.

Root Cause: A required app is failing to install, a dependency is missing, or the ESP timeout is too short for large apps.

Steps: Review which app is blocking, check the app's install status and detection rules, increase the ESP timeout, reduce the list of blocking apps.

Resolution: Fix or remove the failing required app; only mark essential apps as blocking.

Best Practices

- Keep the list of "blocking" apps on the ESP minimal — only what's truly required for first use.
- Use OEM/reseller registration so hardware hashes are uploaded automatically at scale.
- Use a clear device-naming template (e.g., `CORP-%SERIAL%`).
- Test profiles with a pilot dynamic group before broad rollout.
- Prefer User-Driven + Entra Join for standard staff; Self-deploying only for unattended/kiosk devices.

Security Considerations

Autopilot enforces identity and compliance from first boot. Set the user account type to **Standard** (not Administrator) in the profile to follow least privilege. Combine with Conditional Access so the device must be compliant before reaching data. Self-deploying mode requires **TPM 2.0** attestation to prevent rogue devices from auto-provisioning.

PowerShell Commands

```
# Install the community script and capture the hardware hash to CSV
Install-Script -Name Get-WindowsAutopilotInfo -Force
Set-ExecutionPolicy -Scope Process -ExecutionPolicy RemoteSigned
Get-WindowsAutopilotInfo -OutputFile C:\HWID\AutopilotHWID.csv

# Capture and upload directly to Intune (interactive sign-in)
Get-WindowsAutopilotInfo -Online
```

Interview Questions and Answers

BEGINNER **What is Windows Autopilot in one sentence?**

A cloud service that turns a new OEM Windows device into a fully managed corporate device automatically when the user first signs in — no imaging required.

BEGINNER **What is a hardware hash?**

A unique fingerprint of a device collected from its firmware, used to register and identify it in Autopilot.

INTERMEDIATE **Difference between User-Driven and Self-Deploying mode?**

User-Driven requires the user to sign in and enrolls the device under that user. Self-Deploying needs no credentials and provisions automatically — ideal for kiosks — but requires TPM 2.0.

INTERMEDIATE **What does the Enrollment Status Page do?**

It blocks access to the desktop and shows progress while required apps, policies, and certificates install during provisioning.

ADVANCED **A pilot device runs generic OOBE with no branding. How do you diagnose it?**

Confirm the device is registered in Autopilot, the profile is assigned (Profile status = Assigned), the dynamic group has populated, and the device had internet during OOBE. Re-import the hash and reset if needed.

ADVANCED **Why use Pre-provisioning (White Glove)?**

It front-loads the slow part — installing apps and policies — at IT or a reseller, so when the end user receives the device, sign-in to desktop takes only a couple of minutes.

MD-102 Exam Notes

> **Exam Focus**
>
> Memorize the four deployment modes and their use cases. Remember: Self-deploying needs **TPM 2.0**. The ESP can **block** device use until apps install. Autopilot devices are targeted via **dynamic device groups** using the ZTDId rule. Hardware hash registration is the prerequisite step.

Hands-On Lab

> **Lab 2.1 — Build an Autopilot deployment**
>
> 1. On a VM, run `Get-WindowsAutopilotInfo -OutputFile C:\HWID\hash.csv` . 2. Import the CSV under *Windows enrollment → Devices*. 3. Create a dynamic device group using the ZTDId rule. 4. Create a User-Driven deployment profile (Entra joined, Standard user, name template `CORP-%SERIAL%`) and assign it to the group. 5. Configure the ESP to block until apps install. 6. Reset the VM (*Settings → Recovery*) and watch it go through the Autopilot OOBE.

Summary

Autopilot delivers zero-touch, cloud-driven provisioning by registering a device's hardware hash and applying a deployment profile during OOBE. It supports User-Driven, Self-deploying, Pre-provisioning, and existing-device modes, uses dynamic groups for targeting, and uses the Enrollment Status Page to control the first-run experience. It is central to modern endpoint management and the MD-102 exam.

Mini Quiz

1. What must be collected and uploaded to register an Autopilot device?
2. Which mode requires TPM 2.0 and no user sign-in?
3. What targets Autopilot profiles to devices?
4. What screen blocks the desktop until apps install?
5. Which mode pre-loads apps before the user receives the device?
6. Which join type is standard for user-driven corporate laptops?

Answers: 1) The hardware hash (as CSV). 2) Self-deploying mode. 3) A dynamic device group (ZTDId rule). 4) The Enrollment Status Page (ESP). 5) Pre-provisioning (White Glove). 6) Microsoft Entra joined (User-Driven).

3. Provisioning Packages

Introduction

A **provisioning package** (a `.ppkg` file) is a small, self-contained bundle of configuration settings that transforms an existing Windows installation into a corporate-ready device — without re-imaging and without necessarily needing a network. You build it with the **Windows Configuration Designer** (WCD), then apply it by double-clicking it, plugging in a USB stick, or running it during OOBE.

Why This Topic Is Important

Provisioning packages shine where Autopilot can't reach: locations with poor or no internet, air-gapped environments, quick bulk configuration of kiosks, or adding many devices to a domain offline. They're a practical fallback every endpoint admin should understand, and the exam expects you to know when to choose them over Autopilot.

Key Concepts

- **Windows Configuration Designer (WCD):** The tool (from the Microsoft Store / ADK) used to author `.ppkg` files.
- **.ppkg file:** The compiled provisioning package applied to devices.
- **Provisioning vs imaging:** Provisioning layers settings on existing Windows; imaging replaces the whole OS.
- **Bulk Entra token:** A token embedded in a package that lets devices Entra-join in bulk without interactive sign-in.

Architecture / Workflow

1. Admin opens WCD and chooses a wizard (e.g., "Provision desktop devices") or advanced editing.
2. Admin configures settings: device name, Wi-Fi, enrollment/join, apps, certificates, policies.
3. WCD compiles the choices into a `.ppkg` (optionally signed/encrypted).
4. The package is distributed via USB, network share, email, or injected during OOBE.
5. Applying the package configures the device in seconds to minutes.

Microsoft Portal Navigation

Provisioning packages are authored in the desktop **Windows Configuration Designer** app rather than a web portal. Bulk-enrollment tokens for Intune relate to:

```
Intune Admin Center → Devices → Enrollment → Windows → Automatic Enrollment (MDM scope must be set)
```

Step-by-Step Configuration

1. Install **Windows Configuration Designer** from the Microsoft Store.
2. Choose *Provision desktop devices*.

3. Set device name template (e.g., `CORP-%SERIAL%`) and optional product key.
4. Configure Wi-Fi (SSID + credentials) so the device connects automatically.
5. Under enrollment, add a **bulk Entra token** to join devices to Entra ID and auto-enroll in Intune.
6. Add certificates, apps, and any policies needed.
7. Click *Create* to produce the `.ppkg`; optionally encrypt and sign it.
8. Apply it: *Settings → Accounts → Access work or school → Add or remove a provisioning package → Add*, or run during OOBE by pressing the Windows key five times and selecting the package from USB.

Real-World Scenario

Scenario: Remote warehouse, weak internet

A logistics company sets up a warehouse with poor connectivity. Autopilot's heavy cloud download is impractical. IT builds a `.ppkg` with Wi-Fi, device naming, and a bulk Entra token, puts it on a USB stick, and a local technician applies it to 40 devices offline in an afternoon. The devices Entra-join and enroll once they next reach a stronger connection.

Enterprise Deployment Example

A hospital provisions hundreds of identical bedside kiosk PCs. A single signed, encrypted provisioning package configures kiosk mode, locks settings, joins Entra, and enrolls in Intune. Technicians apply it via USB during the initial rollout, achieving consistency without a deployment server.

Troubleshooting

Provisioning package fails to apply

Symptoms: Error when adding the package, or settings don't take effect.

Root Cause: Wrong Windows edition, expired bulk token, unsigned package blocked by policy, or corrupt `.ppkg`.

Steps: Confirm the target edition matches; regenerate the bulk Entra token if expired (tokens have limited validity); rebuild and sign the package; test on a clean device.

Resolution: Re-author with a fresh token and matching settings, then reapply.

Best Practices

- Always **sign and encrypt** packages — they can contain credentials and tokens.
- Remember bulk Entra tokens **expire**; rebuild packages before tokens lapse.
- Use provisioning packages for offline/low-bandwidth sites; prefer Autopilot when good internet exists.
- Keep packages in version control / documented storage.

Security Considerations

A `.ppkg` can embed Wi-Fi passwords, certificates, and join tokens, so treat it as sensitive. Encrypt it, restrict who can run it, and store it securely. An unsigned package could be tampered with, so enforce signing.

PowerShell Commands

```
# Apply a provisioning package via PowerShell
Install-ProvisioningPackage -PackagePath "C:\Packages\Corp.ppkg" -ForceInstall

# List packages already applied to the device
Get-ProvisioningPackage -AllInstalledPackages

# Remove a provisioning package
Remove-ProvisioningPackage -PackageId <GUID>
```

Interview Questions and Answers

BEGINNER **What is a provisioning package?**

A `.ppkg` file that applies configuration (naming, Wi-Fi, join, apps, policies) to an existing Windows install without re-imaging.

INTERMEDIATE **When would you use a provisioning package instead of Autopilot?**

In offline or low-bandwidth locations, air-gapped environments, or for quick bulk configuration where the cloud-driven Autopilot flow isn't practical.

ADVANCED **A package applied last month now fails to Entra-join devices. Likely cause?**

The embedded bulk Entra token has expired. Rebuild the package with a fresh token.

MD-102 Exam Notes

> **Exam Focus**
>
> Provisioning packages = built in **Windows Configuration Designer**, file type **.ppkg**, applied offline via USB or OOBE, can carry a **bulk Entra token** (which expires). Choose them for low/no-connectivity scenarios.

Hands-On Lab

> **Lab 3.1 — Create and apply a package**
>
> 1. Install Windows Configuration Designer. 2. Run "Provision desktop devices." 3. Set a device-name template and Wi-Fi. 4. Add a bulk Entra token. 5. Export the `.ppkg`. 6. On a test VM, apply it via *Settings → Accounts → Access work or school → Add a provisioning package* and verify the device name and join state with `dsregcmd /status`.

Summary

Provisioning packages are lightweight `.ppkg` files built in Windows Configuration Designer that configure existing Windows devices offline — ideal for poor connectivity, kiosks, and bulk setup. They can embed bulk Entra tokens (which expire) and should always be signed and encrypted.

Mini Quiz

1. What tool creates provisioning packages?
2. What is the file extension?
3. What kind of token lets devices Entra-join in bulk, and what's its catch?
4. Name one scenario where a package beats Autopilot.
5. Why should packages be signed and encrypted?

Answers: 1) Windows Configuration Designer (WCD). 2) `.ppkg`. 3) A bulk Entra token; it expires. 4) Offline/low-bandwidth or air-gapped sites. 5) They can contain credentials, Wi-Fi passwords, and join tokens.

4. Subscription Activation

Introduction

Subscription Activation lets a Windows device automatically "step up" from a lower edition (such as Windows Pro) to a higher edition (such as Windows Enterprise) based purely on the **license assigned to the signing-in user** — no product keys, no re-imaging, no reboot in many cases. When a properly licensed user signs in, Windows quietly upgrades the edition; if the license is removed, it steps back down.

Why This Topic Is Important

Enterprises licensed through Microsoft 365 E3/E5 or Windows Enterprise E3/E5 use this to deploy Pro devices from any source and have them become Enterprise automatically when users sign in. It removes key management entirely and is a clean exam topic distinguishing license-based activation from traditional keys.

Key Concepts

- **Step-up:** Moving from Pro to Enterprise (or Education) via license, not media.
- **Entra ID licensing:** The user's Windows Enterprise entitlement (via M365 E3/E5, Windows E3/E5) drives activation.
- **No keys:** No MAK/KMS key entry; activation follows identity.
- **Step-down:** When the entitlement is removed, the device reverts to the base edition at the next renewal.

Architecture / Workflow

1. Device runs a base edition (e.g., Windows 11 Pro) and is Entra-joined.
2. A user with a Windows Enterprise entitlement signs in.
3. Windows contacts Entra ID, confirms the entitlement, and upgrades the edition automatically.
4. If the entitlement is later removed, Windows steps down at renewal.

Microsoft Portal Navigation

```
Microsoft 365 Admin Center → Users → Active users → (select user) → Licenses and apps
```

```
Microsoft Entra Admin Center → Identity → Users → (user) → Licenses
```

```
On device: Settings → System → Activation
```

Step-by-Step Configuration

1. Ensure devices run Windows 10/11 **Pro** (or Pro Education).
2. Assign users a license that includes the Windows Enterprise entitlement (M365 E3/E5 or Windows E3/E5).
3. Ensure devices are **Microsoft Entra joined** (or Hybrid joined).

4. Have the user sign in; verify under *Settings* → *System* → *Activation* that the edition shows Enterprise and is activated.

Real-World Scenario

Scenario: Buy Pro, run Enterprise

A company buys laptops that ship with Windows 11 Pro. Rather than re-imaging to Enterprise or entering keys, IT simply assigns each employee a Microsoft 365 E3 license. When users sign in, their devices auto-upgrade to Windows 11 Enterprise, unlocking advanced security and management features with zero manual activation.

Enterprise Deployment Example

A 10,000-seat enterprise standardizes on Pro hardware from multiple OEMs. By centralizing entitlements in Entra ID via M365 E5, every device becomes Enterprise on first sign-in regardless of vendor. License reclamation (removing a leaver's license) automatically steps their old device back down, simplifying compliance audits.

Troubleshooting

Device not stepping up to Enterprise

Symptoms: Activation still shows Pro after sign-in.

Root Cause: User lacks the Enterprise entitlement, device isn't Entra-joined, base edition is wrong, or sync hasn't completed.

Steps: Confirm the user's license includes Windows Enterprise; confirm `dsregcmd /status` shows AzureAdJoined = YES; confirm the base edition is Pro; allow time/connectivity for activation.

Resolution: Assign the correct license, ensure Entra join, then re-sign-in.

Best Practices

- Deploy devices as **Pro** and let licensing handle Enterprise — avoids key management.
- Reclaim licenses from leavers so devices step down and seats free up.
- Verify activation status as part of onboarding checks.

Security Considerations

Because activation follows identity, protecting accounts (MFA, Conditional Access) directly protects edition entitlements. Removing a compromised or departed user's license automatically reduces the device's capabilities.

PowerShell Commands

```
# Show licensing/activation status
slmgr /dlv

# Show current Windows edition
Get-ComputerInfo | Select-Object WindowsProductName, OsBuildNumber

# Confirm Entra join (required for subscription activation)
dsregcmd /status
```

Interview Questions and Answers

BEGINNER **What does Subscription Activation do?**

It automatically upgrades a device's Windows edition (e.g., Pro → Enterprise) based on the signed-in user's license, with no keys or re-imaging.

INTERMEDIATE **What requirements must be met for it to work?**

The device runs a supported base edition (Pro), is Entra-joined, and the user has a Windows Enterprise entitlement (e.g., M365 E3/E5).

ADVANCED **What happens when a licensed user's entitlement is removed?**

The device steps down to its base edition at the next license renewal/check.

MD-102 Exam Notes

Exam Focus

Subscription Activation = **no product keys**, edition follows the **user's license**, requires **Entra join** and a Windows Enterprise entitlement (M365 E3/E5 or Windows E3/E5). It steps up *and* steps down automatically.

Hands-On Lab

Lab 4.1 — Observe a step-up

1. On an Entra-joined Windows Pro VM, check *Settings → System → Activation* (should read Pro). 2. In M365 Admin Center, assign the test user an E3/E5 license. 3. Sign in as that user. 4. Re-check Activation — it should now show Enterprise. 5. Remove the license and observe step-down at renewal.

Summary

Subscription Activation upgrades Windows editions automatically based on user licensing, eliminating product keys and re-imaging. It needs a Pro base edition, Entra join, and a Windows Enterprise entitlement, and it reverses when the license is removed.

Mini Quiz

1. What drives the edition upgrade in Subscription Activation?
2. What base edition is typically required?

3. Name two licenses that include the Enterprise entitlement.
4. What join type is required?
5. What happens when the license is removed?

Answers: 1) The signed-in user's license/entitlement. 2) Windows Pro. 3) Microsoft 365 E3/E5 or Windows Enterprise E3/E5. 4) Microsoft Entra join (or Hybrid). 5) The device steps down to the base edition at renewal.

5. Microsoft Entra Join

Introduction

Microsoft Entra Join (formerly Azure AD Join) connects a Windows device *directly* to your cloud identity — Microsoft Entra ID — with no dependency on an on-premises Active Directory domain. The device is "owned" by the organization in the cloud, users sign in with their work accounts, and the device can be managed by Intune. This is the foundation of **cloud-native** endpoint management.

Why This Topic Is Important

Entra Join is the default identity model for modern, cloud-first organizations and the target state Microsoft steers everyone toward. It enables single sign-on to Microsoft 365 and SaaS apps, Conditional Access, Windows Hello for Business, and automatic Intune enrollment. The exam contrasts it constantly with Hybrid Join, so you must know exactly what each enables.

Key Concepts

- **Entra registered (Workplace join):** A personal/BYOD device gets a lightweight identity in Entra for access — the org does *not* own it. Used for BYOD + app/MAM access.
- **Entra joined:** A corporate device fully joined to the cloud directory; org owns it; supports full MDM management.
- **Hybrid Entra joined:** Device joined to *both* on-prem AD and Entra ID (covered in Topic 6).
- **Primary Refresh Token (PRT):** A token issued after join/sign-in that powers SSO across apps.
- **Automatic MDM enrollment:** When configured, an Entra-joined device auto-enrolls into Intune at join time.

The three device identity states

State	Ownership	Sign-in	Typical Use
Entra registered	User (BYOD)	Personal account + work access	BYOD phones/PCs, MAM
Entra joined	Organization	Work account	Corporate cloud-native PCs
Hybrid Entra joined	Organization	Work (on-prem AD) account	Orgs with on-prem AD dependencies

Architecture / Workflow

1. During OOBE (or via Settings), the device is set up "for an organization."
2. User authenticates to Entra ID with their work account.
3. Entra registers the device object and establishes trust; a **PRT** is issued for SSO.
4. If automatic enrollment is configured, the device enrolls in Intune (MDM) immediately.
5. Intune applies configuration, compliance, and apps; Conditional Access can now evaluate the device.

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Identity → Devices → All devices
```

```
Microsoft Entra Admin Center → Identity → Devices → Device settings (join permissions, device limits)
```

```
Intune Admin Center → Devices → Enrollment → Windows → Automatic Enrollment (MDM user scope)
```

```
On device: Settings → Accounts → Access work or school → Connect → Join this device to Microsoft Entra ID
```

Step-by-Step Configuration

A. Allow users to join devices

1. Go to *Entra → Devices → Device settings*.
2. Set "Users may join devices to Microsoft Entra ID" to All (or selected groups).
3. Optionally set the maximum number of devices per user.

B. Enable automatic MDM enrollment

1. Go to *Intune → Devices → Enrollment → Windows → Automatic Enrollment*.
2. Set MDM user scope to All (or a group) so Entra-joined devices auto-enroll in Intune.

C. Join a device

1. New device: during OOBE choose "Set up for an organization" and sign in with the work account.
2. Existing device: *Settings → Accounts → Access work or school → Connect → Join this device to Microsoft Entra ID*.
3. Verify with `dsregcmd /status` (AzureAdJoined = YES).

Real-World Scenario

Scenario: Cloud-first startup

A startup with no on-prem servers uses Microsoft 365 for everything. Every laptop is Entra joined and auto-enrolled in Intune. Employees get SSO to Outlook, Teams, and SaaS apps; Conditional Access requires compliant, encrypted devices; and IT manages everything from the cloud with no domain controller anywhere.

Enterprise Deployment Example

A media company migrating off on-prem AD moves new hires straight to Entra Join via Autopilot. Legacy file-share access is replaced by SharePoint/OneDrive, and the few remaining on-prem apps are reached through Entra application proxy or VPN. Over time the domain shrinks as Entra-joined devices become the standard.

Troubleshooting

Device won't Entra-join

Symptoms: "Something went wrong" during join; join option greyed out.

Root Cause: User not allowed to join, device limit reached, MFA/Conditional Access blocking, or time/sync issues.

Steps: Check *Device settings* join permission and per-user device limit; verify the account isn't blocked by CA; confirm internet and correct system time; review `dsregcmd /status` diagnostics.

Resolution: Grant join rights / raise device limit, satisfy MFA, retry.

Joined but not enrolling in Intune

Symptoms: Device is Entra joined but no Intune policies apply.

Root Cause: Automatic enrollment (MDM scope) not set, or no Intune license on the user.

Steps: Confirm MDM user scope includes the user; confirm the user has an Intune license; force a sync.

Resolution: Enable automatic enrollment and assign an Intune license.

Best Practices

- Pair Entra Join with **automatic MDM enrollment** so management is immediate.
- Limit who can join devices and set a sensible per-user device cap.
- Use Conditional Access to require compliant, Entra-joined devices for sensitive apps.
- Default new deployments to Entra Join unless on-prem dependencies force Hybrid.

Security Considerations

Entra Join enables device-based Conditional Access, Windows Hello for Business (passwordless), and PRT-based SSO with strong protections. Restrict join permissions to prevent unmanaged devices entering the directory, and require MFA at join time.

PowerShell Commands

```
# Detailed join + SSO diagnostics (most important command here)
dsregcmd /status

# Show PRT and SSO state
dsregcmd /status | findstr "AzureAdPrt"

# Force a device/MDM sync after join
Get-ScheduledTask | ? {$_.TaskName -like "*PushLaunch*"} | Start-ScheduledTask
```

Interview Questions and Answers

BEGINNER **What is Microsoft Entra Join?**

Joining a Windows device directly to the cloud directory (Entra ID), with no on-prem AD, so it's org-owned and cloud-managed.

BEGINNER **Difference between Entra registered and Entra joined?**

Registered = personal/BYOD device with a light identity for access (user-owned). Joined = corporate device fully owned and managed by the org.

INTERMEDIATE **How does an Entra-joined device end up in Intune automatically?**

By enabling Automatic Enrollment (MDM user scope) in Intune so the device enrolls at join time, provided the user has an Intune license.

ADVANCED **What is a Primary Refresh Token and why does it matter?**

A PRT is issued after join/sign-in and powers seamless SSO across apps and resources; PRT issues often manifest as broken SSO, diagnosable via `dsregcmd /status`.

MD-102 Exam Notes

Exam Focus

Know the three states cold: **registered** (BYOD, user-owned), **joined** (corporate, cloud-only), **hybrid joined** (on-prem AD + Entra). Entra Join + Automatic Enrollment = cloud-native management. `dsregcmd /status` is the go-to diagnostic. Conditional Access can require Entra-joined/compliant devices.

Hands-On Lab

Lab 5.1 — Entra join + auto-enroll

1. In Entra, set device join permission to All. 2. In Intune, set MDM user scope to All. 3. On a Windows VM, go to *Settings → Accounts → Access work or school → Connect → Join this device to Microsoft Entra ID*. 4. Sign in with a licensed test user. 5. Run `dsregcmd /status` and confirm AzureAdJoined = YES. 6. In Entra → Devices and Intune → Devices, confirm the device appears and is managed.

Summary

Microsoft Entra Join connects devices directly to cloud identity with no on-prem AD, enabling SSO, Conditional Access, Windows Hello, and automatic Intune enrollment. It is the cloud-native foundation, distinct from Entra registered (BYOD) and Hybrid Entra Join. `dsregcmd /status` is the essential diagnostic.

Mini Quiz

1. Which identity state is for organization-owned, cloud-only PCs?
2. Which state is for personal BYOD devices?
3. What enables automatic Intune enrollment at join time?

4. What command diagnoses join and SSO status?
5. What token powers SSO after join?

Answers: 1) Entra joined. 2) Entra registered. 3) Automatic Enrollment (MDM user scope) + Intune license. 4) `dsregcmd /status`. 5) The Primary Refresh Token (PRT).

6. Hybrid Microsoft Entra Join

Introduction

Hybrid Microsoft Entra Join is for organizations that still rely on an on-premises Active Directory but also want the cloud benefits of Entra ID and Intune. The device is joined to *both* directories: it remains a member of the on-prem AD domain (so Group Policy and on-prem resources keep working) *and* gets a synchronized identity in Entra ID (so Conditional Access, SSO, and cloud management work). It is a **transition state** on the journey to fully cloud-native.

Why This Topic Is Important

Most established enterprises can't drop on-prem AD overnight — they have file servers, legacy apps, and decades of Group Policy. Hybrid Join lets them modernize gradually. The exam loves to test "on-prem dependency → Hybrid Join" reasoning and the role of **Microsoft Entra Connect**.

Key Concepts

- **Microsoft Entra Connect (Entra Connect Sync):** The tool that synchronizes on-prem AD objects (users, devices) up to Entra ID.
- **Service Connection Point (SCP):** An AD configuration object that tells domain-joined devices which Entra tenant to register with.
- **Device sync:** The on-prem computer object is synced to Entra so the device gains a cloud identity.
- **Co-management:** Managing a device with both Configuration Manager and Intune simultaneously.
- **GPO + MDM coexistence:** Both Group Policy and Intune policies can apply; conflicts need a clear authority (often Intune via MDM precedence).

Architecture / Workflow

1. Device is joined to on-prem AD as normal.
2. **Entra Connect** syncs the on-prem device object to Entra ID, and the SCP points devices to the tenant.
3. The device automatically registers with Entra ID (hybrid join) in the background.
4. Optionally, Group Policy or Intune enrollment rules trigger MDM enrollment into Intune.
5. The device now appears in both AD and Entra; it can use GPO *and* cloud features.

Entra Join vs Hybrid Entra Join — Comparison

Aspect	Entra Join	Hybrid Entra Join
On-prem AD needed	No	Yes
Identity location	Cloud only (Entra)	On-prem AD + Entra
Group Policy	No (use Intune)	Yes (plus Intune)
Best for	Cloud-first, new deployments	On-prem dependencies, gradual migration
Setup complexity	Low	Higher (Entra Connect, SCP)
Enrollment	Auto at join	Via GPO / auto-enroll rules
Sign-in	Work (cloud) account	On-prem AD account (synced)

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Identity → Devices → All devices (join type shows "Hybrid Microsoft Entra joined")
```

```
Microsoft Entra Admin Center → Identity → Hybrid management → Microsoft Entra Connect
```

```
Intune Admin Center → Devices → Enrollment → Windows → Automatic Enrollment (for GPO-driven MDM enrollment)
```

Step-by-Step Configuration (high level)

1. Install and configure **Microsoft Entra Connect** on-prem; enable device synchronization.
2. In Entra Connect, run the *Configure device options → Configure Hybrid Microsoft Entra join* wizard; it creates the **SCP** and sets the target tenant.
3. Ensure devices can reach Entra endpoints (firewall/proxy).
4. Domain-joined devices auto-register to Entra on next sign-in/sync.
5. To manage with Intune, deploy a Group Policy ("Enable automatic MDM enrollment using default Azure AD credentials") and enable automatic enrollment in Intune.
6. Verify with `dsregcmd /status` (AzureAdJoined = YES *and* DomainJoined = YES).

Real-World Scenario

Scenario: Bank with legacy apps

A bank runs critical on-prem applications that authenticate against Active Directory and depend on Group Policy lockdowns. It can't go cloud-only yet. Using Hybrid Entra Join, its domain-joined PCs gain Entra identities, so the bank can enforce Conditional Access and start managing settings via Intune, while existing GPOs and on-prem apps continue working untouched.

Enterprise Deployment Example

A manufacturer with 20,000 domain-joined PCs enables Hybrid Join through Entra Connect and turns on co-management. Workloads (compliance, Windows Update, endpoint protection) are gradually shifted from Configuration Manager to Intune slider by slider, letting the org modernize without a risky big-bang migration.

Troubleshooting

Devices not becoming Hybrid joined

Symptoms: `dsregcmd /status` shows DomainJoined = YES but AzureAdJoined = NO.

Root Cause: Entra Connect not syncing devices, missing/incorrect SCP, network can't reach Entra endpoints, or sync delay.

Steps: Confirm device sync is enabled in Entra Connect; verify the SCP points to the correct tenant; check connectivity to Entra registration endpoints; allow a sync cycle and reboot.

Resolution: Fix Entra Connect device sync / SCP, restore connectivity, re-run registration.

GPO and Intune policy conflict

Symptoms: A setting from Intune doesn't stick; GPO appears to win (or vice versa).

Root Cause: Both authorities target the same setting.

Steps: Identify the conflicting setting; decide a single authority; for co-managed devices move the relevant workload slider to Intune; remove the duplicate GPO setting.

Resolution: Establish one source of truth per setting.

Best Practices

- Use Hybrid Join only when on-prem dependencies genuinely require it — it's a stepping stone, not an end state.
- Keep **Entra Connect** healthy and monitored; device sync is critical.
- Avoid policy conflicts by assigning one authority (GPO *or* Intune) per setting.
- Plan a migration path toward pure Entra Join over time.

Security Considerations

Hybrid Join extends Conditional Access to domain-joined devices, but you now have two attack surfaces (on-prem AD and cloud). Protect Entra Connect (it holds sync privileges), enforce MFA, and ensure on-prem AD is hardened, since its compromise can affect the cloud.

PowerShell Commands

```
# Confirm hybrid state: both should be YES
dsregcmd /status

# Force a device registration attempt
dsregcmd /join   # (re-attempts hybrid join in supported builds)

# On the Entra Connect server, start a sync cycle
Start-ADSyncSyncCycle -PolicyType Delta
```

Interview Questions and Answers

BEGINNER **What is Hybrid Entra Join?**

A device joined to both on-prem AD and Entra ID, so it keeps Group Policy and on-prem access while gaining cloud features.

INTERMEDIATE **What tool synchronizes on-prem identities and enables hybrid join?**

Microsoft Entra Connect, which syncs AD objects to Entra and configures the Service Connection Point.

INTERMEDIATE **When would you pick Hybrid over pure Entra Join?**

When the org has on-prem dependencies — legacy apps, on-prem resource access, or Group Policy requirements — and can't go cloud-only yet.

ADVANCED **A device shows DomainJoined=YES, AzureAdJoined=NO. What's wrong?**

It's domain joined but not registering to Entra — likely Entra Connect device sync is off, the SCP is missing/wrong, or the device can't reach Entra endpoints.

MD-102 Exam Notes

> **Exam Focus**
>
> Hybrid Join = on-prem AD + Entra. Requires **Microsoft Entra Connect** and a **Service Connection Point (SCP)**. MDM enrollment is typically triggered via **Group Policy** ("Enable automatic MDM enrollment"). In `dsregcmd /status` both **DomainJoined** and **AzureAdJoined** read YES. It's a migration/transition state.

Hands-On Lab

> **Lab 6.1 — Verify hybrid state (lab/domain environment)**
>
> 1. On a domain-joined Windows VM, run `dsregcmd /status` and note DomainJoined / AzureAdJoined. 2. On the Entra Connect server, confirm device sync is enabled and run `Start-ADSyncSyncCycle -PolicyType Delta`. 3. Reboot the client and re-check `dsregcmd /status` for AzureAdJoined = YES. 4. Apply the "Enable automatic MDM enrollment" GPO and confirm the device appears in Intune.

Summary

Hybrid Entra Join joins devices to both on-prem AD and Entra ID, preserving Group Policy and on-prem access while unlocking cloud management and Conditional Access. It depends on Microsoft Entra Connect

and an SCP, often uses GPO-driven MDM enrollment, and is a transitional model toward full cloud-native Entra Join.

Mini Quiz

1. Which directories is a hybrid-joined device a member of?
2. What tool enables hybrid join and identity sync?
3. What AD object points devices to the right tenant?
4. In `dsregcmd /status`, what two values are YES for a hybrid device?
5. How is MDM enrollment typically triggered for hybrid devices?

Answers: 1) On-prem AD and Entra ID. 2) Microsoft Entra Connect. 3) The Service Connection Point (SCP). 4) DomainJoined and AzureAdJoined. 5) Group Policy ("Enable automatic MDM enrollment") with automatic enrollment configured.

7. Windows Update Deployment

Introduction

Keeping Windows patched is one of the most important security duties of an endpoint administrator. **Windows Update for Business (WUfB)** lets you control *when* and *how* updates reach managed devices straight from Microsoft's cloud — no on-prem update server required. You group devices into **update rings**, defer updates by a set number of days, and roll changes out in waves to catch problems before they hit everyone.

Why This Topic Is Important

Unpatched devices are a top breach cause, but updating everything at once can break business apps. Administrators must balance security and stability using rings and deferrals. WUfB (plus the newer **Windows Autopatch**) is the modern, cloud-native way to do this and is firmly on the MD-102 exam.

Key Concepts

- **Quality updates:** Monthly security/reliability fixes (the "Patch Tuesday" cumulative updates).
- **Feature updates:** Major OS version upgrades (e.g., Windows 11 23H2 → 24H2).
- **Update rings:** Groups of devices with their own deferral and deadline settings, used for phased rollout (e.g., Pilot → Broad).
- **Deferral period:** Days to wait after release before installing.
- **Deadline & grace period:** When installation/restart becomes mandatory.
- **Feature update profile:** Pins devices to a specific Windows version.
- **Windows Autopatch:** A Microsoft-managed service that automates rings and update orchestration for you.
- **Expedite updates:** Force a critical patch out immediately, bypassing normal deferrals.

Architecture / Workflow

1. Devices get updates directly from Microsoft's Windows Update service.
2. Intune sends **policy** (rings, feature profiles), not the update content itself.
3. Each device honors its ring's deferral and deadline settings.
4. Pilot rings receive updates first; if stable, broad rings follow.
5. Reporting shows update status and failures across the fleet.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Windows → Windows updates → Update rings for Windows 10 and later
```

```
Intune Admin Center → Devices → Windows → Windows updates → Feature updates for Windows 10 and later
```

```
Intune Admin Center → Devices → Windows → Windows updates → Quality updates (Expedite)
```

```
Intune Admin Center → Reports → Windows updates (Windows Update for Business reports)
```

Step-by-Step Configuration

A. Create a Pilot update ring

1. Go to *Windows updates* → *Update rings* → *Create profile*.
2. Name it "Ring 1 - Pilot".
3. Set quality update deferral = 2 days; feature update deferral = 0–7 days.
4. Set a deadline (e.g., 2 days) and grace period (e.g., 1 day) for restarts.
5. Configure user experience (e.g., auto-install with active hours).
6. Assign to a small pilot device group.

B. Create a Broad ring

1. Duplicate the pilot ring; name it "Ring 2 - Broad".
2. Increase quality deferral (e.g., 7–9 days) so it lags the pilot.
3. Assign to the rest of the fleet.

C. Pin a feature version

1. Go to *Feature updates* → *Create profile*.
2. Select the target version (e.g., Windows 11 23H2).
3. Assign to device groups to hold them on that version until you're ready.

D. Expedite a critical patch

1. Go to *Quality updates* → *Create* (Expedite).
2. Select the specific patch and a fast restart setting.
3. Assign to the affected devices to push immediately.

Real-World Scenario

Scenario: Phased rollout saves the day

A company sends a monthly update to "Ring 1 - Pilot" (50 IT and volunteer devices) first. A driver conflict surfaces on the pilot. Because the 950 broad-ring devices have a 7-day deferral, IT pauses the rollout and fixes the driver *before* the broad ring is ever affected — preventing a company-wide outage.

Scenario: Zero-day patch

A critical vulnerability is announced. IT uses **Expedite quality updates** to push the fix to all devices within hours, ignoring normal deferrals, then resumes regular ring schedules afterward.

Enterprise Deployment Example

A 25,000-seat enterprise adopts **Windows Autopatch**. Devices are auto-distributed across Test, First, Fast, and Broad rings; Microsoft orchestrates quality and feature updates, halts rollouts automatically if failure thresholds are exceeded, and provides reporting. IT effort drops dramatically while patch compliance rises.

Troubleshooting

Updates not installing on targeted devices

Symptoms: Devices remain on old patch level past the deadline.

Root Cause: Conflicting update policies, a pause in effect, connectivity issues, or low disk space.

Steps: Check for multiple/conflicting update ring assignments; confirm no pause is active; verify internet and free disk space; review the Windows Update for Business reports and device update status.

Resolution: Resolve policy conflicts, lift pauses, free space, and force a sync.

Feature update won't move past a pinned version

Symptoms: Devices stay on an older Windows version.

Root Cause: A feature update profile is pinning them deliberately.

Steps: Check the assigned feature update profile target version; update or unassign it to allow advancement.

Resolution: Set the profile to the newer target version.

Best Practices

- Always use **multiple rings** (at least Pilot + Broad) for phased rollout.
- Set realistic deadlines and grace periods to balance security with user disruption.
- Use **feature update profiles** to control major version jumps; don't let them happen randomly.
- Reserve **Expedite** for true emergencies.
- Monitor the Windows Update for Business reports regularly.
- Consider **Windows Autopatch** at scale to reduce manual effort.

Security Considerations

Timely patching closes known vulnerabilities — the single biggest preventable risk. Enforce deadlines so users can't defer indefinitely, and pair update compliance with Conditional Access so devices that fall dangerously behind can be flagged non-compliant and lose access until patched.

PowerShell Commands

```
# Check installed update history
Get-HotFix | Sort-Object InstalledOn -Descending | Select-Object -First 10

# Trigger an update scan (using the PSWindowsUpdate module)
Install-Module PSWindowsUpdate -Force
Get-WindowsUpdate
Install-WindowsUpdate -AcceptAll -AutoReboot

# Reset the Windows Update components if stuck
Stop-Service wuauserv, bits; Remove-Item C:\Windows\SoftwareDistribution -Recurse -
Force; Start-Service wuauserv, bits
```

Interview Questions and Answers

BEGINNER **Difference between quality and feature updates?**

Quality updates are monthly security/reliability fixes; feature updates are major OS version upgrades.

BEGINNER **What is an update ring?**

A group of devices with its own deferral/deadline settings, used to roll updates out in phases.

INTERMEDIATE **How do you stop devices from jumping to a new Windows version too early?**

Assign a feature update profile that pins them to a specific target version until you choose to advance it.

INTERMEDIATE **A critical zero-day needs patching immediately. What do you use?**

Expedite quality updates, which pushes a specific patch right away, bypassing normal deferral periods.

ADVANCED **What does Windows Autopatch add over plain WUfB rings?**

Microsoft-managed ring distribution and orchestration, automatic rollout halting on failures, and built-in reporting — reducing manual administration at scale.

MD-102 Exam Notes

Exam Focus

WUfB delivers **policy, not content** (devices still pull updates from Microsoft). Know **quality vs feature** updates, **update rings** + deferrals/deadlines, **feature update profiles** (version pinning), **Expedite** for emergencies, and that **Windows Autopatch** automates the whole ring process.

Hands-On Lab

Lab 7.1 — Build update rings

1. Create "Ring 1 - Pilot" with a 2-day quality deferral and assign to a small group. 2. Create "Ring 2 - Broad" with a 7-day deferral for everyone else. 3. Create a feature update profile pinning Windows 11 23H2. 4. Open *Reports → Windows updates* and review device update status. 5. Create an Expedite quality update for a recent patch and observe targeting.

Summary

Windows Update for Business manages patching from the cloud using update rings, deferrals, deadlines, and feature update profiles, enabling safe phased rollouts. Expedite handles emergencies, reporting tracks compliance, and Windows Autopatch automates the entire process at enterprise scale.

Mini Quiz

1. Which update type is a major OS version upgrade?
2. What groups devices for phased rollout?
3. How do you pin devices to a specific Windows version?
4. What feature pushes a critical patch immediately?
5. What does WUfB actually send to devices — policy or update files?
6. What Microsoft service fully automates update rings?

Answers: 1) Feature update. 2) Update rings. 3) A feature update profile. 4) Expedite quality updates. 5) Policy (devices get the files from Microsoft). 6) Windows Autopatch.

8. Cloud-Based Deployment Strategy

Introduction

This topic ties Section 1 together. **Cloud-based deployment** means provisioning, joining, managing, and updating Windows devices entirely from Microsoft's cloud (Entra ID + Intune) instead of relying on on-prem servers and manual imaging. It is the destination that Autopilot, Entra Join, and Windows Update for Business were all building toward.

Why This Topic Is Important

Organizations adopt cloud deployment to support remote work, cut infrastructure costs, improve security, and scale quickly. As an endpoint administrator you'll be expected to design and defend a cloud deployment strategy — and to know how it differs from the legacy on-prem model.

Key Concepts

- **Cloud-native endpoint:** Entra-joined + Intune-managed, no on-prem dependency.
- **Zero-touch provisioning:** Autopilot delivers ready devices with no IT handling.
- **Co-management:** Bridging Configuration Manager and Intune during migration.
- **Cloud management gateway:** Lets Configuration Manager manage internet-based devices during transition.
- **Single pane of glass:** Managing all endpoints from the Intune Admin Center.

Architecture / Workflow

A full cloud deployment combines the pieces from this section:

1. **Provision:** Autopilot delivers and configures devices (or provisioning packages for offline sites).
2. **Identity:** Entra Join attaches devices to cloud identity (Hybrid Join only if on-prem is required).
3. **Manage:** Intune applies configuration, compliance, and apps automatically.
4. **Secure:** Conditional Access + BitLocker + Defender enforce a healthy state.
5. **Update:** Windows Update for Business / Autopatch keeps everything patched.

Cloud vs On-Prem Deployment — Comparison

Aspect	Cloud-Based (Modern)	On-Prem (Traditional)
Provisioning	Autopilot, zero-touch	MDT/Configuration Manager imaging
Identity	Entra Join	AD domain join
Management	Intune (cloud)	Group Policy + Configuration Manager
Infrastructure	None on-prem	Domain controllers, distribution points
Remote work	Native — works anywhere	Needs VPN/line of sight
Scaling	Elastic, fast	Hardware-bound
Updates	WUfB / Autopatch	WSUS / Configuration Manager

Microsoft Portal Navigation

```
Intune Admin Center → Home (unified endpoint management dashboard)
```

```
Intune Admin Center → Devices → Overview (fleet health, enrollment, compliance)
```

```
Microsoft Entra Admin Center → Identity → Devices → All devices
```

Step-by-Step Configuration (strategy rollout)

1. Set up the tenant: enable automatic enrollment and configure Entra device settings.
2. Stand up Autopilot: register devices and create deployment profiles.
3. Build baseline Intune policies: compliance, configuration, BitLocker, Defender.
4. Configure Windows Update rings.
5. Pilot with a small group, validate, then expand fleet-wide.
6. (If hybrid) enable co-management and shift workloads to Intune over time.

Real-World Scenario

Scenario: Going fully cloud

A 1,200-person company closes its on-prem datacenter. New laptops ship via Autopilot, Entra-join on first sign-in, auto-enroll in Intune, encrypt with BitLocker, and patch via WUfB. IT manages the entire global fleet from the Intune Admin Center with no VPN, no domain controllers, and no imaging lab.

Enterprise Deployment Example

A global firm with offices on five continents migrates from Configuration Manager to Intune using co-management. They enable the cloud management gateway so existing devices stay manageable over the

internet, then move compliance, update, and protection workloads to Intune ring by ring. New devices skip the legacy stack entirely via Autopilot, and within two years the on-prem footprint is decommissioned.

Troubleshooting

Inconsistent management across the fleet

Symptoms: Some devices get cloud policy, others don't; mixed compliance.

Root Cause: Mixed identity states (registered vs joined vs hybrid), missing automatic enrollment, or unassigned policies.

Steps: Audit device join types in Entra; confirm automatic enrollment and licensing; verify policy assignments cover all device groups.

Resolution: Standardize join type, ensure enrollment, and assign baseline policies to all-device groups.

Best Practices

- Aim for **cloud-native** (Entra Join + Intune) as the end state.
- Use co-management as a **bridge**, not a destination.
- Establish baseline policies (compliance, configuration, update, security) before scaling.
- Pilot everything; expand in waves.
- Manage from a single console (Intune Admin Center) for consistency.

Security Considerations

Cloud deployment lets you enforce identity, compliance, encryption, and patching from first boot and anywhere on the internet — a major security upgrade over drift-prone on-prem images. Conditional Access becomes the gatekeeper: only healthy, compliant, Entra-joined devices reach corporate data.

PowerShell Commands

```
# Confirm a device is cloud-native (AzureAdJoined = YES, DomainJoined = NO)
dsregcmd /status

# Connect to Microsoft Graph for cloud management automation
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.Read.All"
Get-MgDeviceManagementManagedDevice | Select-Object DeviceName, ComplianceState,
OperatingSystem
```

Interview Questions and Answers

BEGINNER **What does a cloud-native endpoint look like?**

Entra joined and Intune managed, with no on-prem AD or Configuration Manager dependency.

INTERMEDIATE **What is co-management and why use it?**

Managing a device with both Configuration Manager and Intune at once, used to migrate workloads to the cloud gradually without a risky big-bang cutover.

ADVANCED **Design a cloud deployment for a 2,000-seat remote company.**

Autopilot for provisioning, Entra Join for identity, automatic Intune enrollment, baseline compliance/configuration/BitLocker/Defender policies, WUfB rings for patching, and Conditional Access requiring compliant devices — all managed from the Intune Admin Center, piloted then scaled in waves.

MD-102 Exam Notes

Exam Focus

Cloud-native = Entra Join + Intune, no on-prem. **Co-management** bridges Configuration Manager and Intune during migration; workloads move via sliders. The exam favors cloud-first designs and uses on-prem dependencies as the only justification for hybrid/Configuration Manager.

Hands-On Lab

Lab 8.1 — Map your cloud strategy

1. In Intune → Devices → Overview, review enrollment and compliance numbers. 2. In Entra → Devices, categorize devices by join type. 3. Identify any registered/hybrid devices that should become Entra joined. 4. Draft a one-page rollout plan: provision (Autopilot), identity (Entra Join), manage (Intune baselines), update (WUfB rings), secure (CA + BitLocker + Defender).

Summary

Cloud-based deployment unifies Autopilot, Entra Join, Intune, security, and Windows Update for Business into a single internet-driven model that needs no on-prem infrastructure. Co-management bridges legacy environments during migration, while the end state is fully cloud-native management from the Intune Admin Center.

Mini Quiz

1. What two components define a cloud-native endpoint?
2. What bridges Configuration Manager and Intune during migration?
3. Which tool provisions devices with zero touch?
4. What becomes the access gatekeeper in a cloud model?
5. What's the recommended approach before scaling fleet-wide?

Answers: 1) Entra Join + Intune. 2) Co-management. 3) Windows Autopilot. 4) Conditional Access. 5) Pilot first, then expand in waves.

9. Section 1 — Consolidated Interview Preparation

Introduction

This chapter gathers the most likely interview questions for the "Deploy Windows Client" domain into the categories employers actually use: most-asked, scenario-based, troubleshooting, practical administrator, and production-environment. Practice answering these aloud in two to three sentences.

Most Asked Interview Questions

1. What is Windows Autopilot and what problem does it solve?

Cloud zero-touch provisioning that turns a new OEM device into a managed corporate device on first sign-in, eliminating imaging and letting devices ship straight to users.

2. Explain Entra registered vs Entra joined vs Hybrid Entra joined.

Registered = user-owned BYOD with light identity; Joined = org-owned cloud-only device; Hybrid = device in both on-prem AD and Entra for transition scenarios.

3. What is the Enrollment Status Page?

A screen during provisioning that blocks the desktop while required apps, policies, and certificates install.

4. Difference between quality and feature updates?

Quality = monthly security/reliability fixes; Feature = major OS version upgrades.

5. What does `dsregcmd /status` tell you?

The device's join state (AzureAdJoined, DomainJoined, EnterpriseJoined), PRT/SSO status, and MDM enrollment details.

Scenario-Based Questions

1. A remote employee needs a ready laptop shipped to their home. Design it.

OEM-register the device in Autopilot, assign a User-Driven Entra-join profile to the device's dynamic group, configure the ESP, and let the user self-provision on first sign-in.

2. A site has almost no internet but needs 50 devices configured fast.

Build a signed/encrypted provisioning package with Wi-Fi, naming, and a bulk Entra token, and apply it via USB offline.

3. The company buys Pro laptops but needs Enterprise features.

Use Subscription Activation — assign M365/Windows E3/E5 licenses so Entra-joined devices step up to Enterprise on sign-in.

4. A monthly update broke an app on pilot devices.

Because broad rings have longer deferrals, pause the rollout, fix the app/driver, then resume — the phased ring design contained the impact.

Troubleshooting Questions

1. An Autopilot device shows generic OOBE with no branding. Why?

Not registered, profile unassigned, dynamic group empty, or no internet during OOBE.

2. `dsregcmd` shows DomainJoined=YES, AzureAdJoined=NO. Cause?

Hybrid registration failing — Entra Connect device sync off, missing/wrong SCP, or blocked Entra endpoints.

3. Device is Entra joined but gets no Intune policy. Cause?

Automatic enrollment (MDM scope) not set or the user lacks an Intune license.

Practical Administrator Questions

1. How do you collect a hardware hash?

Run `Get-WindowsAutopilotInfo` to export a CSV (or `-Online` to upload directly).

2. How do you target Autopilot profiles?

With a dynamic device group using the ZTDId membership rule.

3. How do you pin devices to a Windows version?

Assign a feature update profile with the desired target version.

Production Environment Questions

1. How would you migrate 10,000 domain-joined PCs to the cloud safely?

Enable Hybrid Join via Entra Connect, turn on co-management, shift workloads to Intune ring by ring, and route new devices through Autopilot to skip the legacy stack.

2. How do you ensure only healthy devices reach corporate data?

Conditional Access requiring compliant, Entra-joined devices, backed by Intune compliance policies (encryption, patch level, Defender).

3. How do you patch safely without breaking the business?

Multiple update rings with staggered deferrals (Pilot → Broad), deadlines, feature version pinning, and Expedite reserved for emergencies.

Summary

Interviewers for endpoint roles probe Autopilot, the three Entra identity states, enrollment, updates, and the cloud-vs-hybrid decision. Answer with the business reason, the specific Microsoft feature, and the exact tool or portal path.

10. Section 1 — Troubleshooting Reference

Introduction

A consolidated, symptom-first reference for the deployment problems you'll meet most often. Each entry follows Symptoms → Root Cause → Steps → Resolution.

1. Device not enrolling into Intune

Symptoms: Device joined but no policies/apps; absent from Intune.

Root Cause: Automatic enrollment not configured, missing Intune license, or MDM user scope excludes the user.

Steps: Verify MDM user scope (Intune → Enrollment → Automatic Enrollment); confirm Intune license; run `dsregcmd /status`; force sync.

Resolution: Enable automatic enrollment and assign a license; device enrolls on next sync.

2. Autopilot deployment shows generic setup

Symptoms: No corporate branding; personal OOBE.

Root Cause: Unregistered hash, unassigned profile, empty dynamic group, or no internet.

Steps: Confirm device under Autopilot devices; profile status Assigned; group populated; connectivity present.

Resolution: Re-import hash, wait for assignment, reset and re-run OOBE.

3. Enrollment Status Page stuck/timing out

Symptoms: Hangs on installing apps; timeout error.

Root Cause: A blocking required app failing, or ESP timeout too short.

Steps: Identify the failing app and its detection rules; reduce blocking apps; increase timeout.

Resolution: Fix/remove failing app; mark only essentials as blocking.

4. Hybrid join not completing

Symptoms: DomainJoined=YES, AzureAdJoined=NO.

Root Cause: Entra Connect device sync off, SCP missing/wrong, blocked Entra endpoints.

Steps: Enable device sync; verify SCP/tenant; check connectivity; run a delta sync and reboot.

Resolution: Correct Entra Connect/SCP and connectivity; device registers.

5. Provisioning package fails to apply

Symptoms: Error adding package or settings don't take.

Root Cause: Wrong edition, expired bulk token, unsigned/corrupt package.

Steps: Match edition; regenerate expired token; rebuild and sign; test on clean device.

Resolution: Re-author with fresh token and reapply.

6. Subscription Activation not upgrading edition

Symptoms: Activation still shows Pro.

Root Cause: No Enterprise entitlement, not Entra joined, wrong base edition, or sync delay.

Steps: Confirm license includes Windows Enterprise; confirm Entra join; confirm Pro base; allow time.

Resolution: Assign correct license, ensure Entra join, re-sign-in.

7. Windows Update not installing

Symptoms: Devices past deadline still unpatched.

Root Cause: Conflicting ring assignments, active pause, connectivity, or low disk space.

Steps: Resolve conflicting rings; lift pauses; check connectivity/disk; review WUfB reports.

Resolution: Single ring authority, free space, force sync.

8. Device won't Entra-join

Symptoms: Join fails or option greyed out.

Root Cause: Join not permitted, device limit reached, CA/MFA blocking, time skew.

Steps: Check device settings join permission and limit; satisfy MFA; correct time; review `dsregcmd`.

Resolution: Grant rights/raise limit, complete MFA, retry.

Best Practices

- Start every deployment investigation with `dsregcmd /status`.
- Check assignment scope (groups, licenses) before assuming a feature is broken.
- Validate connectivity to Microsoft endpoints during OOBE/registration.
- Keep one authority per setting to avoid GPO/Intune conflicts.

Summary

Most Section 1 deployment failures trace back to a handful of causes: missing enrollment configuration, unassigned profiles/policies, expired tokens, broken sync (Entra Connect/SCP), connectivity, or licensing gaps. A symptom-first approach plus `dsregcmd /status` resolves the majority quickly.

End of Section 1 — Deploy Windows Client.
Next: *Section 2 — Manage Identity and Compliance* (Entra ID, device registration, compliance policies, Conditional Access, authentication methods, passwordless, BitLocker, compliance reporting). Request it whenever you're ready and it will follow the same structure.

MICROSOFT CERTIFICATION

MD-102

Microsoft Endpoint Administrator

Section 2 — Manage Identity and Compliance

A Practical, Interview-Focused Enterprise Study Guide

Microsoft Entra ID · Device Registration · Compliance Policies · Conditional Access

Authentication Methods · Passwordless · BitLocker · Compliance Reporting

Real-World Administration • Troubleshooting • Hands-On Labs • Interview Preparation

Section 2 Contents

1. Microsoft Entra ID
2. Device Registration and Identity
3. Device Compliance Policies
4. Conditional Access
5. Authentication Methods
6. Passwordless Authentication
7. BitLocker Management
8. Compliance Reporting
9. Section 2 — Consolidated Interview Preparation
10. Section 2 — Troubleshooting Reference

How to use this section: Identity and compliance are the heart of Zero Trust endpoint management. Each topic keeps the fixed structure — concepts, architecture, exact portal navigation, step-by-step configuration, scenarios, enterprise examples, troubleshooting, best practices, security, PowerShell, interview Q&A, exam notes, a hands-on lab, summary, and a mini quiz. Build everything in a trial tenant as you read.

1. Microsoft Entra ID

Introduction

Microsoft Entra ID (formerly Azure Active Directory) is Microsoft's cloud-based identity and access management service. It is the directory that stores your users, groups, and devices, and it decides *who* can sign in and *what* they can reach. Every other topic in this section — device registration, compliance, Conditional Access, authentication — sits on top of Entra ID. If on-prem Active Directory was the "phone book and gatekeeper" of the old world, Entra ID is its cloud successor for the Microsoft 365 era.

Why This Topic Is Important

Identity is the new security perimeter. With remote work and SaaS apps, the firewall no longer defines the boundary — the user's identity does. Entra ID is where you control sign-in, enforce MFA, manage groups that target Intune policies, and register devices. An endpoint administrator who doesn't understand Entra ID cannot manage devices effectively, and the exam treats it as foundational knowledge.

Key Concepts

- **Tenant:** Your organization's dedicated, isolated instance of Entra ID (e.g., `contoso.onmicrosoft.com`).
- **User objects:** Cloud-only users, or users synced from on-prem AD via Entra Connect.
- **Groups:** *Security groups* (for access/policy targeting) and *Microsoft 365 groups* (for collaboration). Membership can be *Assigned* (manual) or *Dynamic* (rule-based).
- **Roles (RBAC):** Built-in roles like Global Administrator, Intune Administrator, User Administrator that grant scoped permissions.
- **Licenses:** Entitlements (M365 E3/E5, Entra ID P1/P2) that unlock features such as Conditional Access (P1) and risk-based policies (P2).
- **Device objects:** Records of registered/joined devices, central to compliance and Conditional Access.
- **Entra Connect:** Syncs on-prem AD identities to the cloud for hybrid environments.

Architecture / Workflow

1. Identities live in the **tenant** (cloud-only or synced from AD).
2. Users are organized into **groups**, which are used to target licenses, apps, and Intune policies.
3. When a user signs in, Entra ID **authenticates** them (password, MFA, passwordless).
4. **Conditional Access** then evaluates signals (user, device, location, risk) to allow, block, or challenge.
5. **Roles** govern what administrators can do; **licenses** govern what features are available.

Entra ID license tiers (what unlocks what)

Tier	Key Capabilities
Entra ID Free	Basic user/group management, SSO, security defaults
Entra ID P1	Conditional Access, dynamic groups, self-service password reset, group-based licensing
Entra ID P2	Everything in P1 plus Identity Protection (risk-based policies), Privileged Identity Management (PIM)

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Identity → Users → All users
```

```
Microsoft Entra Admin Center → Identity → Groups → All groups
```

```
Microsoft Entra Admin Center → Identity → Roles & admins
```

```
Microsoft Entra Admin Center → Identity → Overview → (tenant info, licenses)
```

Step-by-Step Configuration

A. Create a dynamic security group (used to target Intune policies)

1. Go to *Identity* → *Groups* → *All groups* → *New group*.
2. Group type = Security; Membership type = **Dynamic Device** (or Dynamic User).
3. Click *Add dynamic query* and build a rule, e.g., all Windows devices.
4. Save; Entra evaluates the rule and populates membership automatically.

```
# Example dynamic device rule: all Windows devices
(device.deviceOSType -eq "Windows")

# Example dynamic user rule: all users in the Sales department
(user.department -eq "Sales")
```

B. Assign an administrative role

1. Go to *Identity* → *Roles & admins*.
2. Select a role (e.g., **Intune Administrator**) → Add assignments.
3. Pick the user; for privileged roles, prefer eligible (PIM) over permanent.

Real-World Scenario

Scenario: Targeting by department

The Finance team needs a stricter device configuration. Instead of manually maintaining a list, the admin creates a **dynamic user group** with the rule `user.department -eq "Finance"`. As HR updates each new hire's department attribute, the group membership updates automatically and the right Intune policies apply with no manual effort.

Enterprise Deployment Example

A 15,000-employee company drives all access and management from Entra ID. Users sync from on-prem AD via Entra Connect, dynamic groups segment the workforce by region and department, Entra ID P2 enables risk-based Conditional Access, and PIM grants just-in-time admin access so no one holds standing Global Admin rights. Group-based licensing automatically assigns M365 E5 to anyone in the "Employees" group.

Troubleshooting

User not getting expected policy or license

Symptoms: A user lacks an app, license, or Intune policy others have.

Root Cause: They're not in the targeting group, a dynamic rule attribute is blank, or group processing hasn't completed.

Steps: Check group membership; verify the attribute the dynamic rule relies on (e.g., department) is populated; allow time for dynamic evaluation.

Resolution: Correct the user attribute or add them to the group; membership and assignments follow.

Admin can't perform an action

Symptoms: "You don't have permission" errors in a portal.

Root Cause: Missing or insufficient role, or an eligible PIM role not yet activated.

Steps: Review the user's role assignments; if using PIM, activate the eligible role; confirm the role scope covers the object.

Resolution: Assign/activate the correct least-privilege role.

Best Practices

- Use **dynamic groups** to target policies so membership stays accurate automatically.
- Apply **least privilege** — assign the narrowest role that works (e.g., Intune Administrator, not Global Admin).
- Use **PIM** for just-in-time privileged access where licensed (P2).
- Keep emergency "break-glass" accounts excluded from Conditional Access.
- Keep user attributes clean — dynamic rules depend on them.

Security Considerations

Entra ID is the identity perimeter, so protect it fiercely: enforce MFA for all admins, minimize standing privilege with PIM, monitor sign-in and audit logs, and maintain two cloud-only break-glass accounts with strong protection in case lockout or CA misconfiguration occurs.

PowerShell Commands

```
# Connect to Microsoft Graph
Connect-MgGraph -Scopes "User.Read.All","Group.Read.All"

# List users
Get-MgUser -All | Select-Object DisplayName, UserPrincipalName

# List groups and membership
Get-MgGroup -All | Select-Object DisplayName, GroupTypes
Get-MgGroupMember -GroupId <group-id>

# View role assignments
Get-MgRoleManagementDirectoryRoleAssignment -All
```

Interview Questions and Answers

BEGINNER **What is Microsoft Entra ID?**

Microsoft's cloud identity and access management service (formerly Azure AD) that stores users, groups, and devices and controls authentication and access.

BEGINNER **Difference between a security group and a Microsoft 365 group?**

Security groups manage access and target policies/licenses; Microsoft 365 groups are for collaboration (shared mailbox, SharePoint, Teams).

INTERMEDIATE **What's the difference between assigned and dynamic group membership?**

Assigned membership is manually maintained; dynamic membership is automatically built from a rule based on user or device attributes.

INTERMEDIATE **What license tier unlocks Conditional Access?**

Entra ID P1 (included in M365 E3); P2 adds risk-based policies and PIM.

ADVANCED **How do you reduce the risk of standing administrative privilege?**

Use Privileged Identity Management (PIM, P2) to make privileged roles eligible and activated just-in-time with approval, instead of permanently assigned.

MD-102 Exam Notes

> **Exam Focus**
>
> Entra ID is the foundation. Know **group types** (security vs M365) and **membership types** (assigned vs dynamic), license tiers (**P1 = Conditional Access**, **P2 = Identity Protection + PIM**), least-privilege roles like **Intune Administrator**, and that dynamic groups are the standard way to target Intune policies.

Hands-On Lab

Lab 1.1 — Build identity scaffolding

1. In Entra → Users, create two test users with different department attributes. 2. Create a dynamic user group with rule `user.department -eq "Sales"` . 3. Create a dynamic device group for all Windows devices. 4. Assign one user the Intune Administrator role. 5. Confirm dynamic membership populates correctly.

Summary

Microsoft Entra ID is the cloud identity service underpinning all endpoint management. It stores users, groups, and devices; uses dynamic groups to target policies; controls access through roles and licenses; and serves as the security perimeter that authentication, compliance, and Conditional Access build upon.

Mini Quiz

1. What was Entra ID formerly called?
2. Which membership type updates automatically from a rule?
3. Which license tier unlocks Conditional Access?
4. Which P2 feature provides just-in-time admin access?
5. What kind of group is used to target Intune policies?

Answers: 1) Azure Active Directory (Azure AD). 2) Dynamic. 3) Entra ID P1. 4) Privileged Identity Management (PIM). 5) A security group (typically dynamic).

2. Device Registration and Identity

Introduction

Device registration is the process of giving a device an identity in Microsoft Entra ID so the organization can recognize it, evaluate its health, and control its access. A device with no identity is invisible to Conditional Access and compliance. This topic builds on the join types from Section 1 (Entra registered, Entra joined, Hybrid joined) and focuses on *how* devices get their identity and *why* that identity matters for security.

Why This Topic Is Important

You cannot enforce "only compliant devices can access email" unless devices have identities to evaluate. Device registration is the bridge between a piece of hardware and your security policies. In interviews you'll be asked to explain the registration states and to diagnose why a device isn't recognized — both common real-world tasks.

Key Concepts

- **Device identity:** The device object in Entra ID representing a physical/virtual machine.
- **Entra registered:** Lightweight identity for personal/BYOD devices; user-owned; enables access and MAM.
- **Entra joined:** Full organizational identity; org-owned; supports full MDM.
- **Hybrid Entra joined:** Identity in both on-prem AD and Entra (transition state).
- **MDM enrollment:** Registering the device with Intune so policies and apps can be delivered. Registration (identity) and enrollment (management) are related but distinct.
- **Device ownership:** Corporate vs Personal — affects which actions and policies apply.

Architecture / Workflow

1. A device authenticates a user or is set up for the organization.
2. Entra creates a **device object** and records its join type and ownership.
3. If configured, the device **enrolls in Intune** (MDM) for management.
4. Intune evaluates **compliance** and reports device state back to Entra.
5. **Conditional Access** uses the device identity + compliance to make access decisions.

Registration vs enrollment

Term	What it gives	Where
Registration / Join	A device *identity* in Entra ID	Microsoft Entra ID
Enrollment	Device *management* (policies, apps)	Microsoft Intune (MDM)

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Identity → Devices → All devices
```

```
Microsoft Entra Admin Center → Identity → Devices → Device settings
```

```
Intune Admin Center → Devices → All devices
```

```
Intune Admin Center → Devices → Enrollment → Windows → Automatic Enrollment
```

Step-by-Step Configuration

A. Control device registration settings

1. Go to *Entra → Devices → Device settings*.
2. Set who may join devices (All / Selected / None).
3. Set the maximum number of devices per user.
4. Require MFA to register or join devices (recommended).

B. Register a personal (BYOD) device

1. On the device, open *Settings → Accounts → Access work or school → Connect*.
2. Enter the work account (do *not* tick "join this device to Entra ID" for BYOD — that's for corporate join).
3. The device becomes **Entra registered**; verify under Entra → Devices.

C. Mark devices as corporate

1. Go to *Intune → Devices → Enrollment → Corporate device identifiers*.
2. Add identifiers (IMEI or serial numbers) so matching devices are tagged Corporate automatically.

Real-World Scenario

Scenario: BYOD email access

An employee wants email on a personal phone. IT doesn't want to fully manage personal hardware, so the device is **Entra registered** and protected with app protection policies (MAM). Conditional Access allows Outlook only from registered devices with MAM in force, keeping corporate data contained without taking over the personal device.

Enterprise Deployment Example

A company segments its fleet: corporate laptops are Entra joined and fully MDM-managed; contractor phones are Entra registered with MAM only. Corporate device identifiers (serial numbers from procurement) are uploaded so new laptops are auto-tagged Corporate, while anything else defaults to Personal — driving different policy sets and Conditional Access rules.

Troubleshooting

Device missing from Entra / not recognized by Conditional Access

Symptoms: CA blocks a device for not being registered/compliant though the user expects access.

Root Cause: Device never registered, registration failed, or stale/duplicate device object.

Steps: Run `dsregcmd /status`; check Entra → Devices for the object; look for duplicates; verify join permission and MFA at registration.

Resolution: Re-register the device, remove stale duplicates, ensure registration is permitted.

Device shows wrong ownership (Personal vs Corporate)

Symptoms: Corporate device treated as Personal, getting the wrong policies.

Root Cause: No corporate device identifier match, or enrolled via a personal path.

Steps: Add the serial/IMEI to corporate device identifiers; re-enroll if needed; or change ownership in Intune.

Resolution: Tag as corporate so correct policies apply.

Best Practices

- Require **MFA** for device registration/join.
- Set a sensible **per-user device limit**.
- Use **corporate device identifiers** so company hardware is tagged correctly and automatically.
- Regularly clean up **stale device objects** in Entra.
- Use Entra registered + MAM for BYOD; Entra joined + MDM for corporate.

Security Considerations

Device identity is what lets Conditional Access distinguish a trusted laptop from an unknown machine. Unmanaged or stale device objects weaken this, so restrict who can register, require MFA, and prune inactive devices. For BYOD, prefer app-level (MAM) protection so you secure data without over-reaching into personal devices.

PowerShell Commands

```
# Device join/registration diagnostics
dsregcmd /status

# List Entra devices via Graph
Connect-MgGraph -Scopes "Device.Read.All"
Get-MgDevice -All | Select-Object DisplayName, TrustType, IsManaged, OperatingSystem

# Find stale devices (no recent sign-in)
Get-MgDevice -All | Where-Object { $_.ApproximateLastSignInDateTime -lt (Get-
Date).AddDays(-90) }
```

Interview Questions and Answers

BEGINNER **What is device registration?**

Giving a device an identity in Entra ID so the organization can recognize and evaluate it for access decisions.

BEGINNER **Difference between registration and enrollment?**

Registration gives the device an identity in Entra; enrollment puts it under Intune management so policies and apps can be applied.

INTERMEDIATE **Which registration state suits BYOD and why?**

Entra registered, because it gives the device a light identity for access while keeping it user-owned, typically paired with MAM rather than full MDM.

ADVANCED **A corporate laptop is being treated as a personal device. How do you fix it?**

Add its serial/IMEI to corporate device identifiers (or change ownership in Intune) so it's tagged Corporate and receives the correct policy set.

MD-102 Exam Notes

Exam Focus

Distinguish **registration/join (identity in Entra)** from **enrollment (management in Intune)**. Know the three identity states and their ownership. **Corporate device identifiers** (serial/IMEI) auto-tag company hardware. BYOD → Entra registered + MAM; corporate → Entra joined + MDM.

Hands-On Lab

Lab 2.1 — Register and classify devices

1. In Entra → Devices → Device settings, require MFA to join and set a device limit. 2. Entra-register a test device via Access work or school. 3. In Intune → Enrollment → Corporate device identifiers, add a serial number. 4. Confirm the device appears in Entra → Devices with the correct join type and ownership.

Summary

Device registration gives hardware an identity in Entra ID, the prerequisite for compliance and Conditional Access. Registration (identity) differs from enrollment (management). BYOD devices are typically Entra registered with MAM, corporate devices Entra joined with MDM, and corporate device identifiers ensure company hardware is classified correctly.

Mini Quiz

1. What does device registration give a device?
2. Which service provides device management (enrollment)?
3. Which state fits BYOD?
4. What auto-tags corporate hardware?
5. What command checks a device's registration state?

Answers: 1) An identity in Entra ID. 2) Microsoft Intune. 3) Entra registered. 4) Corporate device identifiers (serial/IMEI). 5) `dsregcmd /status`.

3. Device Compliance Policies

Introduction

A **device compliance policy** is a set of rules in Intune that defines what a "healthy" device looks like — for example, encryption on, a minimum OS version, a PIN required, and not jailbroken. Intune continuously checks devices against these rules and labels each one **Compliant** or **Not compliant**. That status is then used by **Conditional Access** to allow or block access. Compliance policies are the "health check" at the centre of Zero Trust endpoint security.

Why This Topic Is Important

Compliance policies turn security requirements into something measurable and enforceable. Without them, you have no way to require that a device is encrypted or patched before it touches corporate data. This is one of the most heavily used and heavily tested features — almost every Intune deployment has compliance policies, and interviewers expect you to explain how they integrate with Conditional Access.

Key Concepts

- **Compliance rules:** The conditions a device must meet (encryption, OS version, password, firewall, Defender on, etc.).
- **Compliance status:** Compliant / Not compliant / In grace period / Not evaluated.
- **Actions for noncompliance:** What happens when a device fails — e.g., mark noncompliant after a grace period, send email, remotely lock, or retire.
- **Grace period:** Time given to a failing device to fix itself before being marked noncompliant.
- **Mark devices with no compliance policy as:** A tenant-wide default (Compliant or Not compliant) for devices not targeted by any policy — best set to **Not compliant** for Zero Trust.
- **Compliance + Conditional Access:** The compliance result feeds the "require compliant device" CA control.

Architecture / Workflow

1. Admin creates a compliance policy with rules and noncompliance actions.
2. The policy is assigned to a device or user group.
3. The Intune agent on the device evaluates the rules and reports status.
4. Status is recorded in Intune and surfaced to Entra ID.
5. **Conditional Access** reads the status: compliant devices get access, noncompliant devices are blocked or limited.
6. Noncompliance actions (email, grace period, lock) execute as configured.

Compliance Policy vs Conditional Access — Comparison

Aspect	Compliance Policy	Conditional Access
Purpose	Defines what a healthy device is	Decides access based on conditions
Service	Microsoft Intune	Microsoft Entra ID
Output	Compliant / Not compliant status	Allow / Block / Challenge
Acts on	The device's health	User + device + location + risk
Relationship	Produces the signal	Consumes the signal

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Compliance → Policies
```

```
Intune Admin Center → Devices → Compliance → Settings (mark devices with no policy as...)
```

```
Intune Admin Center → Devices → Compliance → Notifications (email templates)
```

```
Intune Admin Center → Devices → Monitor → Device compliance
```

Step-by-Step Configuration

A. Create a Windows compliance policy

1. Go to *Devices* → *Compliance* → *Policies* → *Create policy*; Platform = Windows 10 and later.
2. Configure rules: require BitLocker, require Secure Boot, minimum OS version, password/PIN length, require Microsoft Defender Antivirus, real-time protection on.
3. Set **Actions for noncompliance**: e.g., "Mark device noncompliant" after a 1-day grace period, plus "Send email to end user."
4. Assign to a device group.

B. Set the tenant-wide compliance default

1. Go to *Compliance* → *Settings*.
2. Set "Mark devices with no compliance policy assigned as" = **Not compliant** (Zero Trust posture).

C. Configure a noncompliance notification

1. Go to *Compliance* → *Notifications* → *Create notification*; design the email.
2. Reference it in the policy's noncompliance actions.

Real-World Scenario

Scenario: Encryption enforcement

A company requires every laptop to be BitLocker-encrypted before accessing SharePoint. A compliance policy requires BitLocker; a laptop with encryption off is marked **Not compliant**. Conditional Access blocks it from SharePoint and the user receives an automated email explaining how to enable encryption. Once compliant, access is restored automatically.

Scenario: Grace period prevents lockout storms

After tightening the minimum OS version, hundreds of devices would instantly fail. A 7-day grace period gives users time to update before being marked noncompliant, avoiding a flood of blocked users on day one.

Enterprise Deployment Example

A regulated enterprise builds tiered compliance: a baseline policy for all devices (encryption, Defender, OS version) and stricter policies for Finance and Executive groups (shorter idle lock, jailbreak detection, firewall required). The tenant default is Not compliant, and Conditional Access requires compliant devices for all M365 apps, so unmanaged or unhealthy devices simply cannot reach data.

Troubleshooting

Compliance policy not applying

Symptoms: Device shows "Not evaluated" or no policy.

Root Cause: Policy not assigned to the device's group, device not enrolled, or hasn't synced.

Steps: Verify assignment groups; confirm the device is enrolled in Intune; force a sync; check the platform matches.

Resolution: Correct assignment/enrollment; status evaluates on next sync.

Device unexpectedly marked noncompliant

Symptoms: A healthy-looking device is blocked by CA.

Root Cause: A specific rule failing (e.g., Defender off, OS below minimum, encryption pending), or grace period expired.

Steps: Open the device's compliance detail in Intune to see *which* setting failed; remediate that setting on the device.

Resolution: Fix the failing setting; device returns to Compliant on re-evaluation.

Best Practices

- Set the tenant default to **Not compliant** for Zero Trust.
- Use **grace periods** to avoid mass lockouts when tightening rules.

- Start with a baseline policy, then layer stricter policies for sensitive groups.
- Pair compliance with **Conditional Access** — a policy with no CA enforcement only reports, it doesn't block.
- Configure user notifications so people know how to self-remediate.

Security Considerations

Compliance policies are only as strong as the Conditional Access that enforces them. A device marked noncompliant changes nothing unless CA requires compliance for access. Always design the two together. Require encryption, current patches, and active threat protection as baseline rules, and treat unmanaged devices as noncompliant by default.

PowerShell Commands

```
# List compliance policies and device compliance state via Graph
Connect-MgGraph -Scopes
"DeviceManagementConfiguration.Read.All","DeviceManagementManagedDevices.Read.All"
Get-MgDeviceManagementDeviceCompliancePolicy | Select-Object DisplayName

# Show compliance state per managed device
Get-MgDeviceManagementManagedDevice | Select-Object DeviceName, ComplianceState,
OSVersion
```

Interview Questions and Answers

BEGINNER **What is a device compliance policy?**

A set of Intune rules defining a healthy device (encryption, OS version, PIN, etc.); Intune marks devices Compliant or Not compliant against it.

BEGINNER **What is a grace period?**

Time given to a failing device to fix itself before being marked noncompliant, preventing immediate lockout.

INTERMEDIATE **How do compliance policies and Conditional Access work together?**

Compliance produces the Compliant/Not compliant signal; Conditional Access consumes it to allow or block access (e.g., "require a compliant device").

INTERMEDIATE **Why set the tenant default to "Not compliant"?**

So devices not covered by any policy are treated as untrusted by default — a Zero Trust posture that prevents unmanaged devices from gaining access.

ADVANCED **A device is blocked but looks healthy. How do you diagnose it?**

Open the device's compliance detail in Intune to find the exact failing setting (e.g., Defender off, OS below minimum, encryption pending), remediate it, and let it re-evaluate.

MD-102 Exam Notes

Exam Focus

Compliance policy (Intune) *produces* the status; Conditional Access (Entra) *enforces* it. Know **actions for noncompliance**, **grace periods**, and the tenant setting **"mark devices with no policy as Not compliant"**. A compliance policy alone doesn't block access — CA does.

Hands-On Lab

Lab 3.1 — Build and test compliance

1. Create a Windows compliance policy requiring BitLocker and a minimum OS version. 2. Add a noncompliance action: mark noncompliant after 1 day + send email. 3. Set the tenant default to Not compliant. 4. Assign to a test group. 5. In Devices → Monitor → Device compliance, confirm status. 6. Toggle a setting on a test device and watch the status change.

Summary

Device compliance policies define and measure device health in Intune, producing a Compliant/Not compliant status that Conditional Access uses to gate access. Grace periods prevent mass lockouts, noncompliance actions notify users, and the tenant default should be Not compliant for Zero Trust. Compliance and Conditional Access must be designed together.

Mini Quiz

1. Which service evaluates compliance?
2. Which service enforces access based on compliance?
3. What prevents mass lockouts when rules tighten?
4. What should the tenant default be for Zero Trust?
5. Does a compliance policy by itself block access?

Answers: 1) Microsoft Intune. 2) Conditional Access (Entra ID). 3) A grace period. 4) Not compliant. 5) No — Conditional Access must enforce it.

4. Conditional Access

Introduction

Conditional Access (CA) is the policy engine of Microsoft Entra ID that enforces access decisions in real time. Think of it as a series of **if-then statements**: *if* a user, device, location, or risk condition is met, *then* require something (MFA, a compliant device) or block access entirely. Conditional Access is the practical enforcement layer of Zero Trust — it's where "only healthy, verified access reaches our data" actually happens.

Why This Topic Is Important

Conditional Access is arguably the single most important security control in Microsoft 365 and one of the most tested MD-102 areas. It's what makes compliance policies, MFA, and device identity *matter*, because it turns those signals into enforced decisions. Every endpoint administrator must be able to design, read, and troubleshoot CA policies.

Key Concepts

- **Assignments (the "if"):** Users/groups, target resources (cloud apps), and conditions (device platform, location, client app, sign-in risk).
- **Access controls (the "then"):** *Grant* controls (block, require MFA, require compliant device, require Entra-joined device, require app protection policy) and *Session* controls (app-enforced restrictions, sign-in frequency).
- **Named locations:** Trusted IP ranges or countries used as conditions.
- **Report-only mode:** Evaluates a policy and logs what *would* happen without enforcing it — essential for safe testing.
- **Sign-in / user risk:** Risk signals from Identity Protection (P2) used as conditions.
- **Break-glass accounts:** Emergency admin accounts excluded from CA to prevent total lockout.

Architecture / Workflow

1. A user attempts to sign in to a resource (e.g., Outlook).
2. Entra ID gathers **signals**: who, what app, what device, where, what risk.
3. All applicable CA policies are evaluated together.
4. The combined result applies the strictest necessary **grant/session controls**.
5. Access is granted (possibly after MFA), limited, or blocked.

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Protection → Conditional Access → Policies
```

```
Microsoft Entra Admin Center → Protection → Conditional Access → Named locations
```

```
Microsoft Entra Admin Center → Protection → Conditional Access → Insights and reporting (report-only)
```

Step-by-Step Configuration

A. Require a compliant device for Microsoft 365

1. Go to *Protection* → *Conditional Access* → *Policies* → *New policy*.
2. Name: "Require compliant device - M365".
3. Assignments → Users: All users (exclude break-glass accounts).
4. Target resources: select the Microsoft 365 apps.
5. Grant: **Require device to be marked as compliant** (and/or Entra hybrid joined).
6. Enable policy in **Report-only** first; review insights; then turn On.

B. Require MFA for all users

1. New policy → Users: All users (exclude break-glass).
2. Target resources: All cloud apps.
3. Grant: **Require multifactor authentication**.
4. Test in report-only, then enable.

C. Block legacy authentication

1. New policy → Conditions → Client apps: select legacy/other clients.
2. Grant: **Block access**.
3. Enable (legacy auth bypasses MFA, so blocking it is a top recommendation).

Real-World Scenario

Scenario: Compliant-device gate

A company requires that email and SharePoint are reachable only from compliant, managed devices. A CA policy grants access only if the device is marked compliant. An employee on an unmanaged home PC is blocked, but the same employee on their managed laptop gets straight in — enforcing security without manual intervention.

Scenario: Risk-based step-up

With Identity Protection (P2), a sign-in from an unusual country with a leaked-credential signal is flagged high risk. A CA policy forces MFA (or blocks) for high-risk sign-ins, stopping an attacker even with a valid password.

Enterprise Deployment Example

A global enterprise runs a layered CA framework: baseline MFA for everyone, block legacy auth, require compliant devices for all M365 apps, require app protection policies for mobile, restrict admin portals to

trusted locations + phishing-resistant MFA, and risk-based policies for elevated sign-in/user risk. Break-glass accounts are excluded and monitored. Every new policy ships in report-only first.

Troubleshooting

Conditional Access blocking legitimate users

Symptoms: Users can't sign in; "access blocked by Conditional Access."

Root Cause: Overly broad policy, device not compliant/registered, or a misconfigured condition.

Steps: Use *Entra → Sign-in logs → Conditional Access tab* for the failed sign-in to see exactly which policy and control blocked it; check device compliance/registration.

Resolution: Adjust the policy scope/exclusions or remediate the device; re-test in report-only.

Risk of total admin lockout

Symptoms: A new CA policy could lock out all admins.

Root Cause: No excluded emergency access accounts.

Steps: Maintain two cloud-only break-glass accounts excluded from all CA policies; test new policies in report-only.

Resolution: Use a break-glass account to sign in and correct the policy.

Best Practices

- Always test new policies in **report-only mode** first.
- Maintain and exclude **break-glass accounts** from all CA policies.
- **Block legacy authentication** — it bypasses MFA.
- Require MFA broadly and compliant devices for sensitive resources.
- Use the **sign-in logs CA tab** as your primary troubleshooting tool.
- Keep policies simple and well-named; fewer, clearer policies are easier to reason about.

Security Considerations

Conditional Access is the enforcement heart of Zero Trust. Misconfiguration can either expose data (too loose) or lock out the entire org (too strict) — hence report-only testing and break-glass accounts are mandatory. Blocking legacy auth and requiring phishing-resistant MFA for admins are among the highest-impact security moves you can make.

PowerShell Commands

```
# List Conditional Access policies via Graph
Connect-MgGraph -Scopes "Policy.Read.All"
Get-MgIdentityConditionalAccessPolicy | Select-Object DisplayName, State

# Review a specific policy's grant controls
(Get-MgIdentityConditionalAccessPolicy -ConditionalAccessPolicyId <id>).GrantControls
```

Interview Questions and Answers

BEGINNER **What is Conditional Access in simple terms?**

An if-then policy engine in Entra ID: if certain conditions are met, then require something (like MFA or a compliant device) or block access.

BEGINNER **Name two common grant controls.**

Require multifactor authentication and require the device to be marked compliant (also: block access, require Entra-joined device, require app protection policy).

INTERMEDIATE **Why test policies in report-only mode?**

To see what a policy *would* do across real sign-ins without enforcing it, avoiding accidental lockouts before going live.

INTERMEDIATE **Why block legacy authentication?**

Legacy protocols can't perform MFA, so they're a common bypass for attackers; blocking them closes that gap.

ADVANCED **How do you safely deploy a strict CA policy to all users?**

Exclude break-glass accounts, enable it in report-only, review the sign-in logs CA insights, then switch to On — with the break-glass account available to recover if needed.

MD-102 Exam Notes

Exam Focus

CA = **assignments (if)** + **access controls (then)**. Know grant controls (block, MFA, compliant device, Entra-joined, app protection) and session controls. **Report-only** for testing, **break-glass** exclusions, and **block legacy auth** are recurring exam themes. Sign-in logs (CA tab) is the troubleshooting tool. Risk-based conditions need **Entra ID P2**.

Hands-On Lab

Lab 4.1 — Conditional Access fundamentals

1. Create a break-glass account and exclude it everywhere. 2. Create a "Require MFA - All users" policy in report-only. 3. Create a "Require compliant device - M365" policy in report-only. 4. Create a "Block legacy authentication" policy. 5. Sign in as a test user and review *Sign-in logs → Conditional Access* to see which policies applied. 6. Promote one policy from report-only to On.

Summary

Conditional Access enforces access in real time using if-then logic over users, apps, devices, location, and risk. It consumes compliance and identity signals to require MFA, demand compliant devices, or block access. Report-only testing, break-glass exclusions, and blocking legacy auth are essential practices, and the sign-in logs CA tab is the key troubleshooting tool.

Mini Quiz

1. What two halves make up a CA policy?
2. Which mode lets you test without enforcing?
3. What accounts must be excluded to avoid lockout?
4. Why block legacy authentication?
5. Where do you see which policy blocked a sign-in?
6. What license is needed for risk-based conditions?

Answers: 1) Assignments (if) and access controls (then). 2) Report-only mode. 3) Break-glass / emergency access accounts. 4) It bypasses MFA. 5) Entra sign-in logs (Conditional Access tab). 6) Entra ID P2.

5. Authentication Methods

Introduction

Authentication methods are the ways users prove who they are when signing in — passwords, the Microsoft Authenticator app, FIDO2 security keys, Windows Hello for Business, SMS/voice codes, and more. Administrators choose which methods are allowed, who can use them, and which are required for multifactor authentication (MFA). Strengthening authentication is one of the fastest ways to dramatically improve security.

Why This Topic Is Important

Stolen passwords cause the majority of breaches. Choosing strong methods — and moving toward phishing-resistant ones — is the core of identity defense. The MD-102 exam expects you to know the available methods, how to manage them centrally, and how MFA and registration policies work.

Key Concepts

- **Multifactor authentication (MFA):** Requiring two or more factors — something you know (password), have (phone/key), or are (biometric).
- **Authentication methods policy:** The modern, centralized place in Entra to enable/disable methods and scope them to groups.
- **Microsoft Authenticator:** App offering push approval, one-time codes, and passwordless phone sign-in.
- **FIDO2 security keys:** Phishing-resistant hardware keys (USB/NFC).
- **Windows Hello for Business:** Biometric/PIN passwordless sign-in tied to the device (covered more under passwordless).
- **Number matching & context:** Authenticator features that defeat MFA-fatigue attacks.
- **SSPR:** Self-service password reset, letting users reset passwords using registered methods.
- **Authentication strength:** A CA control specifying which method combinations satisfy a policy (e.g., phishing-resistant only).

Strength of common methods

Method	Strength	Notes
Password only	Weak	Phishable, reused, guessable
SMS / voice OTP	Low–medium	Better than password; SIM-swap risk
Authenticator push + number matching	Strong	Resists MFA fatigue
Windows Hello for Business	Very strong	Phishing-resistant, device-bound
FIDO2 security key	Very strong	Phishing-resistant hardware

Architecture / Workflow

1. Admin enables methods in the **Authentication methods policy** and scopes them to groups.
2. Users **register** methods (often nudged via the registration campaign or required at first sign-in).
3. At sign-in, Entra requests the primary factor, then a second factor if MFA is required.
4. **Conditional Access** (with optional authentication strength) decides when MFA or phishing-resistant methods are demanded.
5. Registered methods also power **SSPR**.

Microsoft Portal Navigation

```
Microsoft Entra Admin Center → Protection → Authentication methods → Policies
```

```
Microsoft Entra Admin Center → Protection → Authentication methods → Registration campaign
```

```
Microsoft Entra Admin Center → Protection → Password reset (SSPR)
```

```
Microsoft Entra Admin Center → Protection → Authentication methods → Authentication strengths
```

Step-by-Step Configuration

A. Enable and scope methods

1. Go to *Protection → Authentication methods → Policies*.
2. Enable **Microsoft Authenticator**; set push with number matching and additional context to All users.
3. Optionally enable **FIDO2 security keys** for high-security groups.
4. Disable weak/legacy methods over time as adoption grows.

B. Turn on the registration campaign

1. Go to *Authentication methods → Registration campaign*.
2. Enable it to nudge users from SMS toward the Authenticator app.

C. Configure SSPR

1. Go to *Protection → Password reset*.
2. Enable for a pilot group; require a number of registered methods; enable writeback if hybrid.

Real-World Scenario

> **Scenario: Killing MFA fatigue**
>
> Attackers spam push prompts hoping a tired user taps "Approve." The admin enables **number matching** and additional context in Microsoft Authenticator, so users must type a number shown on the sign-in screen. Blind approvals stop working, and the fatigue attack fails.

Scenario: Self-service password reset

A user forgets their password at 2 a.m. Because SSPR is enabled and they've registered the Authenticator app, they reset it themselves in minutes — no help-desk ticket, no downtime.

Enterprise Deployment Example

An enterprise rolls out MFA in waves: enable Authenticator with number matching for everyone, run the registration campaign to migrate users off SMS, deploy FIDO2 keys to administrators and high-risk roles, enable SSPR with password writeback for the synced workforce, and use authentication-strength CA policies to require phishing-resistant methods for admin portals.

Troubleshooting

User can't complete MFA

Symptoms: No push arrives, or the method isn't available.

Root Cause: Method not registered, app notifications disabled, time skew on the device, or method disabled in policy.

Steps: Confirm the user has a registered method; check the Authenticator app notifications/ connectivity; verify the method is enabled and scoped to the user.

Resolution: Re-register the method or enable it; for lockouts, use SSPR or admin reset of methods.

SSPR not working for some users

Symptoms: Reset option missing or fails for hybrid users.

Root Cause: User not in SSPR scope, too few registered methods, or password writeback not configured.

Steps: Confirm SSPR scope and required methods; for hybrid, verify password writeback in Entra Connect.

Resolution: Add to scope / enable writeback; ensure registration.

Best Practices

- Enable **number matching** and context for Authenticator to stop MFA fatigue.
- Move users **off SMS** toward app-based and phishing-resistant methods.
- Use the **Authentication methods policy** (modern) rather than legacy per-user MFA.
- Deploy **FIDO2/Windows Hello** for admins and high-value accounts.
- Enable **SSPR** to cut help-desk load and downtime.

Security Considerations

Not all factors are equal: SMS is better than nothing but vulnerable to SIM-swap and phishing, while FIDO2 and Windows Hello are phishing-resistant. Use authentication strength in Conditional Access to demand the strongest methods where it matters most (admins, sensitive apps), and continually retire weaker methods.

PowerShell Commands

```
# View a user's registered authentication methods via Graph
Connect-MgGraph -Scopes "UserAuthenticationMethod.Read.All"
Get-MgUserAuthenticationMethod -UserId user@contoso.com

# Report registration status across the tenant
Get-MgReportAuthenticationMethodUserRegistrationDetail -All |
  Select-Object UserPrincipalName, IsMfaRegistered, IsSsprRegistered
```

Interview Questions and Answers

BEGINNER **What is MFA?**

Requiring two or more factors to sign in — something you know, have, or are — so a stolen password alone isn't enough.

BEGINNER **Name a phishing-resistant method.**

FIDO2 security keys or Windows Hello for Business.

INTERMEDIATE **What is number matching and what does it prevent?**

A feature where the user types a number shown on the sign-in screen into Authenticator, preventing MFA-fatigue (push-spam) attacks.

INTERMEDIATE **What does SSPR do?**

Lets users reset their own passwords using registered methods, reducing help-desk tickets and downtime; hybrid setups need password writeback.

ADVANCED **How do you require only phishing-resistant MFA for administrators?**

Create a Conditional Access policy targeting admin roles/apps with an authentication strength that allows only phishing-resistant methods (FIDO2 / Windows Hello).

MD-102 Exam Notes

Exam Focus

Use the modern **Authentication methods policy**. Know method strengths (SMS weak; Authenticator + number matching strong; FIDO2/Windows Hello phishing-resistant). **Number matching** defeats MFA fatigue. **SSPR** needs registered methods (and writeback for hybrid). **Authentication strength** is a CA control for demanding specific methods.

Hands-On Lab

Lab 5.1 — Strengthen authentication

1. In Authentication methods → Policies, enable Microsoft Authenticator with number matching for All users. 2. Enable the registration campaign. 3. Enable SSPR for a pilot group requiring two methods. 4. Register the Authenticator app as a test user. 5. Create a CA authentication-strength policy requiring phishing-resistant MFA for an admin group.

Summary

Authentication methods determine how users prove identity. Administrators centrally enable and scope methods, push number matching to beat MFA fatigue, move users off SMS toward phishing-resistant options like FIDO2 and Windows Hello, and enable SSPR to reduce downtime. Authentication strength in Conditional Access enforces the strongest methods where needed.

Mini Quiz

1. What are the three categories of authentication factors?
2. Which feature stops MFA-fatigue push spam?
3. Name two phishing-resistant methods.
4. What does SSPR require for hybrid users?
5. What CA control demands specific method combinations?

Answers: 1) Something you know, have, and are. 2) Number matching. 3) FIDO2 security keys and Windows Hello for Business. 4) Password writeback (via Entra Connect). 5) Authentication strength.

6. Passwordless Authentication

Introduction

Passwordless authentication removes the password from the sign-in entirely, replacing it with something far harder to steal: a biometric (face/fingerprint) or PIN tied to a specific device, a phone-based approval, or a hardware security key. The three pillars in the Microsoft world are **Windows Hello for Business**, **FIDO2 security keys**, and **passwordless phone sign-in** with Microsoft Authenticator. Going passwordless is both more secure *and* more convenient.

Why This Topic Is Important

Passwords are the weakest link — phished, reused, and cracked constantly. Passwordless methods are **phishing-resistant** because there's no shared secret to steal. Microsoft pushes passwordless heavily, and MD-102 expects you to understand the methods, how Windows Hello works, and how to deploy them.

Key Concepts

- **Windows Hello for Business (WHfB):** Device-bound passwordless sign-in using biometrics or a PIN, backed by asymmetric keys and TPM.
- **PIN vs password:** The WHfB PIN never leaves the device and is useless elsewhere, unlike a password that works from anywhere.
- **TPM (Trusted Platform Module):** A secure chip that stores the private key so it can't be extracted.
- **FIDO2 / WebAuthn:** Open standards for phishing-resistant hardware-key sign-in.
- **Passwordless phone sign-in:** Approving sign-in from Microsoft Authenticator without typing a password.
- **Deployment models for WHfB:** Cloud (Entra) trust, key trust, and certificate trust (cloud trust is simplest for cloud-native).

Architecture / Workflow (Windows Hello for Business)

1. The user enrolls a gesture (face, fingerprint, or PIN) on their device.
2. The device generates an **asymmetric key pair**; the private key is protected by the **TPM** and never leaves the device.
3. The public key is registered with Entra ID (and/or AD).
4. At sign-in, the gesture unlocks the private key locally to prove identity — no password travels the network.
5. Because the credential is device-bound and gesture-protected, it can't be phished or replayed.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Enrollment → Windows → Windows Hello for Business
```

```
Intune Admin Center → Endpoint security → Account protection (Windows Hello / credential policies)
```

```
Microsoft Entra Admin Center → Protection → Authentication methods → Policies (FIDO2, passwordless phone sign-in)
```

Step-by-Step Configuration

A. Enable Windows Hello for Business at enrollment

1. Go to *Intune → Devices → Enrollment → Windows → Windows Hello for Business*.
2. Set "Configure Windows Hello for Business" = Enabled.
3. Set minimum PIN length, require TPM, and enable biometrics.
4. This applies tenant-wide at enrollment (or use a targeted Account protection policy for groups).

B. Enable FIDO2 security keys

1. Go to *Entra → Protection → Authentication methods → Policies → FIDO2 security key*.
2. Enable and scope to a group; configure key restrictions if needed.
3. Users register keys at the security info page.

C. Enable passwordless phone sign-in

1. In the Microsoft Authenticator method settings, allow passwordless phone sign-in.
2. Users enable it in the Authenticator app.

Real-World Scenario

Scenario: Phishing-proof executives

After a phishing incident, a company issues **FIDO2 keys** to executives and admins. Even when a convincing fake login page captures a password attempt, there's no password to capture and the key won't authenticate to the fraudulent domain — the attack simply fails.

Scenario: Fast, secure laptop sign-in

Employees sign in to Entra-joined laptops with a fingerprint via Windows Hello. It's faster than typing a password and, because the credential is device-bound and TPM-protected, it can't be reused if stolen.

Enterprise Deployment Example

A cloud-native enterprise standardizes on Windows Hello for Business (cloud trust) for all Entra-joined laptops, issues FIDO2 keys to privileged and shared-device users, and enables passwordless phone sign-in for mobile staff. Conditional Access authentication-strength policies then *require* phishing-resistant methods for admin portals, effectively eliminating passwords from sensitive flows.

Troubleshooting

Windows Hello won't set up

Symptoms: User can't enroll a PIN/biometric; "can't set up" error.

Root Cause: Missing/disabled TPM, WHfB policy not applied, or device not properly joined.

Steps: Confirm TPM present/enabled in firmware; verify the WHfB policy targets the device; confirm Entra join with `dsregcmd /status`.

Resolution: Enable TPM, ensure policy assignment and join, then re-enroll the gesture.

FIDO2 key not accepted

Symptoms: Sign-in won't accept the security key.

Root Cause: FIDO2 method not enabled/scoped, key not registered, or unsupported key.

Steps: Confirm FIDO2 is enabled and the user is in scope; verify the key is registered in security info; check key model support.

Resolution: Enable/scope the method and register the key.

Best Practices

- Prefer **cloud trust** for Windows Hello in cloud-native environments — it's simplest.
- Require **TPM** so credentials are hardware-protected.
- Issue **FIDO2 keys** to admins, privileged, and shared-device users.
- Combine passwordless with CA **authentication strength** to enforce phishing-resistant sign-in where it matters.
- Educate users that the WHfB PIN is device-bound and safer than a password.

Security Considerations

Passwordless methods are phishing-resistant because there's no shared secret to steal and credentials are bound to a device or key. The TPM prevents key extraction. The biggest practical risks are weak fallback methods (don't leave SMS as an easy bypass) and recovery processes — plan secure account recovery so passwordless doesn't create lockouts.

PowerShell Commands

```
# Check Windows Hello for Business status on a device
dsregcmd /status   # look for the "Ngc" (Next Generation Credential) section

# Check TPM presence and readiness
Get-Tpm

# View a user's registered (incl. passwordless) methods via Graph
Get-MgUserAuthenticationMethod -UserId user@contoso.com
```

Interview Questions and Answers

BEGINNER **What is passwordless authentication?**

Signing in without a password, using a device-bound biometric/PIN, a phone approval, or a hardware security key.

BEGINNER **Why is a Windows Hello PIN safer than a password?**

The PIN never leaves the device and only unlocks a TPM-protected key locally, so it's useless to an attacker elsewhere — unlike a password that works from anywhere.

INTERMEDIATE **What role does the TPM play in Windows Hello for Business?**

It securely stores the private key so it can't be extracted, binding the credential to the physical device.

INTERMEDIATE **Name the three Microsoft passwordless options.**

Windows Hello for Business, FIDO2 security keys, and passwordless phone sign-in with Microsoft Authenticator.

ADVANCED **Why is FIDO2 resistant to phishing?**

There's no shared secret to steal, and the key cryptographically binds to the legitimate domain, so it won't authenticate to a fake site even if the user is tricked.

MD-102 Exam Notes

Exam Focus

Three passwordless pillars: **Windows Hello for Business**, **FIDO2 keys**, **passwordless phone sign-in**. WHfB is **device-bound**, uses **TPM-protected asymmetric keys**, and the PIN is local-only. Cloud trust is the simplest WHfB model for Entra-joined devices. Passwordless = phishing-resistant.

Hands-On Lab

Lab 6.1 — Go passwordless

1. In Intune → Enrollment → Windows Hello for Business, enable it with TPM required and biometrics on. 2. On an Entra-joined VM/device, set up a Hello PIN/biometric. 3. Run `dsregcmd /status` and find the Ngc section. 4. In Entra → Authentication methods, enable FIDO2 for a test group and register a key. 5. Enable passwordless phone sign-in in Authenticator.

Summary

Passwordless authentication replaces passwords with phishing-resistant, device-bound or hardware-based credentials: Windows Hello for Business, FIDO2 keys, and passwordless phone sign-in. The TPM protects WHfB keys, the PIN stays local, and combining passwordless with Conditional Access authentication strength removes passwords from the most sensitive sign-ins.

Mini Quiz

1. Name the three Microsoft passwordless methods.
2. What hardware chip protects the Windows Hello key?

3. Why is the Hello PIN safer than a password?
4. What makes FIDO2 phishing-resistant?
5. Which WHfB trust model is simplest for cloud-native devices?

Answers: 1) Windows Hello for Business, FIDO2 security keys, passwordless phone sign-in. 2) The TPM. 3) It's device-bound and never leaves the device. 4) No shared secret + cryptographic binding to the legitimate domain. 5) Cloud (Entra) trust.

7. BitLocker Management

Introduction

BitLocker is the full-disk encryption feature built into Windows. It scrambles everything on the drive so that if a laptop is lost or stolen, the data is unreadable without the proper key. With Intune you can **enforce, monitor, and recover** BitLocker across the entire fleet, and recovery keys are safely **escrowed** (backed up) into Entra ID so no one is ever locked out of their own data.

Why This Topic Is Important

Lost and stolen devices are a constant reality. Encryption is often a legal/compliance requirement (data protection regulations) and a baseline security control. Knowing how to deploy BitLocker via Intune, escrow keys, and run a recovery is core endpoint-admin work and a guaranteed exam and interview topic.

Key Concepts

- **Full-disk encryption:** The whole volume is encrypted, not just individual files.
- **TPM:** Stores the encryption key securely and ties it to the device hardware.
- **Recovery key:** A 48-digit key used to unlock the drive if normal unlock fails; escrowed to Entra ID.
- **Key escrow:** Automatically backing up the recovery key to Entra so admins/users can retrieve it.
- **Encryption method:** e.g., XTS-AES 128/256-bit.
- **Silent enablement:** Encrypting automatically without prompting the user (when prerequisites like TPM are met).
- **Disk encryption policy:** The Intune endpoint-security policy that configures BitLocker.

Architecture / Workflow

1. An Intune **disk encryption policy** targets devices with BitLocker settings.
2. On a TPM-equipped device, BitLocker enables (often silently) and encrypts the drive.
3. The **recovery key is escrowed** to Entra ID automatically.
4. A **compliance policy** can require encryption; Conditional Access blocks unencrypted devices.
5. If unlock fails, an admin or the user retrieves the recovery key from Entra to regain access.

BitLocker vs EFS — Comparison

Aspect	BitLocker	EFS (Encrypting File System)
Scope	Entire volume/disk	Individual files/folders
Protects against	Lost/stolen device, offline attack	Other users on the same OS
Key protection	TPM + recovery key	User certificate/key
Management	Centralized via Intune, key escrow	Per-user, harder to manage centrally
Typical use	Standard device encryption	Niche per-file protection

Microsoft Portal Navigation

```
Intune Admin Center → Endpoint security → Disk encryption → Create policy (BitLocker)
```

```
Intune Admin Center → Devices → Monitor → Encryption report
```

```
Microsoft Entra Admin Center → Identity → Devices → (select device) → BitLocker keys
```

```
User self-service: account portal → Devices → (device) → View BitLocker keys
```

Step-by-Step Configuration

A. Create a BitLocker disk encryption policy

1. Go to *Endpoint security* → *Disk encryption* → *Create policy*; Platform = Windows, Profile = BitLocker.
2. Require encryption of OS and fixed drives.
3. Set encryption method (e.g., XTS-AES 256).
4. Enable **silent enablement** (don't prompt) and require key escrow to Entra ID before encryption.
5. Assign to a device group.

B. Require encryption in compliance

1. In the Windows compliance policy, require BitLocker.
2. Pair with a Conditional Access policy requiring compliant devices.

C. Retrieve a recovery key

1. Go to *Entra* → *Devices* → *(device)* → *BitLocker keys* (or the user's account portal).
2. Copy the recovery key and provide it to the user to unlock the drive.

Real-World Scenario

Scenario: Stolen laptop

An employee's laptop is stolen at an airport. Because BitLocker encrypted the drive and the key is escrowed, the thief sees only unreadable data — no corporate information is exposed. IT marks the device for wipe/retire and issues a replacement; the recovery key was never on the device for the thief to find.

Scenario: Recovery prompt after firmware change

A BIOS update changes the boot measurements and a laptop boots into the BitLocker recovery screen. The user calls the help desk, who retrieve the 48-digit recovery key from Entra, the user types it in, and access is restored in minutes.

Enterprise Deployment Example

A 12,000-device enterprise deploys a single BitLocker policy with silent enablement and mandatory Entra key escrow. The encryption report tracks rollout; compliance requires BitLocker, and Conditional Access blocks any unencrypted device from M365. Help-desk staff have a delegated role to read BitLocker keys for recovery, audited via Entra logs.

Troubleshooting

BitLocker not encrypting

Symptoms: Devices stay unencrypted despite the policy.

Root Cause: No/disabled TPM, policy not assigned, prerequisites unmet, or escrow failing so silent enablement is blocked.

Steps: Check TPM with `Get-Tpm`; confirm policy assignment; review the encryption report for the specific error; verify key escrow connectivity.

Resolution: Enable TPM, fix assignment/escrow; encryption proceeds.

Recovery key not found in Entra

Symptoms: Device prompts for recovery but no key is in Entra.

Root Cause: Encryption happened before escrow was configured, or escrow failed.

Steps: Check if the key exists locally (`manage-bde -protectors -get C:`); force a key backup to Entra; ensure escrow is required going forward.

Resolution: Back up the existing key to Entra and enforce escrow on all devices.

Best Practices

- Always **escrow recovery keys** to Entra ID — and require escrow *before* encryption.
- Use **silent enablement** on TPM devices for a seamless rollout.
- Require encryption in **compliance** and enforce via **Conditional Access**.

- Monitor the **encryption report** for failures.
- Delegate a least-privilege role for help-desk key retrieval and audit access.

Security Considerations

BitLocker protects data at rest against device theft and offline attacks. The recovery key is sensitive — restrict who can read it, audit access, and never store it on the device itself. Combine BitLocker with Secure Boot and TPM for strongest protection, and ensure keys are escrowed so encryption never causes permanent data loss.

PowerShell Commands

```
# Check BitLocker status of all volumes
Get-BitLockerVolume

# Check TPM readiness
Get-Tpm

# Enable BitLocker on C: with TPM protector
Enable-BitLocker -MountPoint "C:" -EncryptionMethod XtsAes256 -TpmProtector

# View recovery key protectors locally
manage-bde -protectors -get C:

# Back up the recovery key to Entra ID
BackupToAAD-BitLockerKeyProtector -MountPoint "C:" -KeyProtectorId <id>
```

Interview Questions and Answers

BEGINNER **What does BitLocker do?**

Encrypts the entire disk so data is unreadable without the key if the device is lost or stolen.

BEGINNER **What is a recovery key?**

A 48-digit key used to unlock the drive when normal unlock fails; it's escrowed to Entra ID for safe retrieval.

INTERMEDIATE **Difference between BitLocker and EFS?**

BitLocker encrypts the whole volume to protect against device theft; EFS encrypts individual files to protect against other users on the same system.

INTERMEDIATE **What does silent enablement mean?**

BitLocker encrypts automatically without prompting the user, provided prerequisites like TPM and key escrow are met.

ADVANCED **A device shows the recovery screen after a firmware update. Why and how do you resolve it?**

Boot measurements changed, triggering recovery. Retrieve the recovery key from Entra (or the user portal), enter it to unlock, and the device resumes normal operation.

MD-102 Exam Notes

Exam Focus

BitLocker is configured via the Intune **Disk encryption (endpoint security)** policy. Know **key escrow to Entra ID**, **silent enablement** (needs TPM), the **48-digit recovery key**, and that BitLocker = full disk while EFS = individual files. Require encryption in compliance and enforce with Conditional Access.

Hands-On Lab

Lab 7.1 — Deploy and recover BitLocker

1. Create a Disk encryption (BitLocker) policy with silent enablement and required Entra escrow; assign it. 2. On a TPM-enabled device, confirm encryption with `Get-BitLockerVolume`. 3. Open Devices → Monitor → Encryption report. 4. In Entra → Devices → (device) → BitLocker keys, view the escrowed recovery key. 5. Practice retrieving the key as if recovering a locked device.

Summary

BitLocker provides full-disk encryption managed centrally through Intune's disk encryption policy, with recovery keys escrowed to Entra ID and silent enablement on TPM devices. It protects against device theft, is enforced via compliance plus Conditional Access, and differs from EFS, which encrypts individual files. Key escrow ensures encryption never causes permanent lockout.

Mini Quiz

1. What does BitLocker encrypt?
2. Where are recovery keys escrowed?
3. How many digits is a recovery key?
4. What chip secures the encryption key?
5. What's the difference between BitLocker and EFS?

Answers: 1) The entire disk/volume. 2) Microsoft Entra ID. 3) 48 digits. 4) The TPM. 5) BitLocker = full disk; EFS = individual files.

8. Compliance Reporting

Introduction

Compliance reporting is how you *see* the security state of your fleet: which devices are compliant, which are failing and why, encryption coverage, and Conditional Access outcomes. Intune and Entra ID provide dashboards, per-device drill-downs, and exportable reports. Good administrators don't just set policies — they **monitor and act on the reports** to keep the environment healthy and to prove compliance to auditors.

Why This Topic Is Important

Policies are worthless if you can't verify they're working. Reporting lets you catch failing devices, demonstrate regulatory compliance, and prioritize remediation. Interviewers ask where to look when something is wrong, and the exam expects familiarity with Intune's monitoring and reporting locations.

Key Concepts

- **Device compliance report:** Fleet-wide compliant/noncompliant counts and per-device status.
- **Per-setting status:** Which specific rule a device failed.
- **Encryption report:** BitLocker readiness and status across devices.
- **Sign-in logs (Entra):** Show Conditional Access outcomes per sign-in.
- **Noncompliance notifications:** Automated emails driving user self-remediation.
- **Organizational / operational / historical reports:** Intune's report categories for summaries, real-time data, and trends.
- **Export & Graph:** Pulling report data via export or the Graph API for dashboards.

Architecture / Workflow

1. Devices report their state to Intune at each **check-in/sync**.
2. Intune aggregates results into **reports and dashboards**.
3. Entra records **sign-in logs** with Conditional Access results.
4. Admins review reports, identify failing devices, and drill into **per-setting** detail.
5. Notifications and remediation actions close the loop; trends are tracked over time.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Monitor → Device compliance
```

```
Intune Admin Center → Devices → Monitor → Noncompliant devices
```

```
Intune Admin Center → Devices → Monitor → Encryption report
```

```
Intune Admin Center → Reports → Device compliance / Device management
```

```
Microsoft Entra Admin Center → Monitoring & health → Sign-in logs (Conditional Access tab)
```

Step-by-Step Configuration

A. Review fleet compliance

1. Go to *Devices* → *Monitor* → *Device compliance* for the overall compliant vs noncompliant breakdown.
2. Open *Noncompliant devices* to get the targeted list.
3. Click a device to see **which setting** failed.

B. Check encryption coverage

1. Go to *Devices* → *Monitor* → *Encryption report*.
2. Identify devices not encrypted or with errors; drill into the reason.

C. Investigate access outcomes

1. Go to *Entra* → *Sign-in logs*; open a sign-in; check the **Conditional Access** tab.
2. See which policies applied and whether access was granted, blocked, or challenged.

D. Export for dashboards

1. Use *Reports* export, or pull data via Microsoft Graph for Power BI dashboards.

Real-World Scenario

Scenario: Monday compliance sweep

Each Monday the admin opens the device compliance report and sees 30 noncompliant laptops. Drilling in, most failed the minimum OS rule. The admin confirms the update rings are pushing the patch, and the automated noncompliance email has already prompted users to update — by Wednesday the count is near zero.

Scenario: Audit evidence

An auditor asks for proof that all laptops are encrypted. The admin exports the encryption report showing BitLocker status across the fleet, providing instant documented evidence.

Enterprise Deployment Example

A large enterprise pulls Intune and Entra data via Microsoft Graph into a Power BI compliance dashboard refreshed daily, giving leadership a single view of encryption coverage, compliance rate, patch level, and CA block trends. Automated noncompliance notifications drive self-remediation, and a weekly operational review targets the most common failing settings.

Troubleshooting

Report shows stale or "Not evaluated" data

Symptoms: Device status looks outdated or never evaluated.

Root Cause: Device hasn't checked in/synced, or was recently enrolled.

Steps: Confirm the device is online and syncing; force a sync from the device or Intune; allow time for the next check-in.

Resolution: After sync, the report reflects current state.

Can't tell why a device is noncompliant

Symptoms: Device flagged noncompliant with no obvious reason.

Root Cause: Not drilling into per-setting detail.

Steps: Open the device in the compliance report and view the per-setting status to see the exact failing rule.

Resolution: Remediate the specific setting identified.

Best Practices

- Review compliance and encryption reports on a **regular cadence**.
- Always drill to **per-setting** detail to find the real cause.
- Use **noncompliance notifications** to push self-remediation.
- Export to **Power BI / Graph** for leadership dashboards and audit evidence.
- Use Entra **sign-in logs** to correlate compliance with access outcomes.

Security Considerations

Reporting is your early-warning system: a rising noncompliance trend can signal a failing policy, a patch problem, or an attack. Treat reports as operational signals, restrict who can view sensitive data (like BitLocker keys), and retain logs/exports as evidence for compliance audits and incident investigations.

PowerShell Commands

```
# Pull compliance state across all managed devices via Graph
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.Read.All"
Get-MgDeviceManagementManagedDevice -All |
  Select-Object DeviceName, ComplianceState, OSVersion, LastSyncDateTime

# Count compliant vs noncompliant
Get-MgDeviceManagementManagedDevice -All |
  Group-Object ComplianceState | Select-Object Name, Count
```

Interview Questions and Answers

BEGINNER **Where do you check overall device compliance?**

Intune → Devices → Monitor → Device compliance (and the Reports section).

BEGINNER **Where do you see if a device is encrypted?**

The encryption report under Devices → Monitor → Encryption report.

INTERMEDIATE **A device is noncompliant — how do you find the exact cause?**

Open the device in the compliance report and view the per-setting status to identify which rule failed.

INTERMEDIATE **Where do you see whether Conditional Access blocked a sign-in?**

In Entra sign-in logs, on the Conditional Access tab of the specific sign-in.

ADVANCED **How would you give leadership a live compliance view?**

Export Intune/Entra data via Microsoft Graph into a Power BI dashboard refreshed on a schedule, surfacing compliance rate, encryption coverage, and CA trends.

MD-102 Exam Notes

Exam Focus

Know the report locations: **Devices** → **Monitor** (device compliance, noncompliant devices, encryption report), **Reports** hub, and Entra **sign-in logs** for Conditional Access outcomes. Per-setting status reveals *why* a device failed. Data can be exported / pulled via **Graph** for dashboards. Stale data usually means the device hasn't synced.

Hands-On Lab

Lab 8.1 — Monitor and report

1. Open Devices → Monitor → Device compliance and note the breakdown. 2. Open Noncompliant devices and drill into one device's per-setting status. 3. Open the Encryption report and identify any unencrypted devices. 4. In Entra → Sign-in logs, inspect the Conditional Access tab of a test sign-in. 5. Export a compliance report (or pull it via Graph).

Summary

Compliance reporting reveals the real security state of the fleet through Intune's compliance and encryption reports, per-setting drill-downs, and Entra sign-in logs for Conditional Access outcomes. Administrators monitor on a cadence, remediate the exact failing settings, drive self-remediation via notifications, and export to Power BI/Graph for dashboards and audit evidence.

Mini Quiz

1. Which report shows BitLocker coverage?
2. How do you find exactly why a device is noncompliant?
3. Where do you see Conditional Access outcomes?
4. What usually causes stale/"Not evaluated" report data?

5. What API pulls report data for dashboards?

Answers: 1) The encryption report. 2) The per-setting status in the device's compliance detail. 3) Entra sign-in logs (Conditional Access tab). 4) The device hasn't checked in/synced. 5) Microsoft Graph.

9. Section 2 — Consolidated Interview Preparation

Introduction

Identity and compliance questions dominate endpoint-admin interviews because they're where security actually lives. Below are the most likely questions grouped the way employers ask them. Answer each in two or three sentences: state the concept, the Microsoft feature, and the exact tool or portal path.

Most Asked Interview Questions

1. What is Conditional Access and how does it work?

An Entra if-then policy engine: if conditions (user, device, location, risk) are met, then enforce controls (MFA, compliant device, block). It consumes compliance and identity signals to make real-time access decisions.

2. How do compliance policies and Conditional Access relate?

Compliance (Intune) produces the Compliant/Not compliant signal; Conditional Access (Entra) enforces it. A compliance policy alone doesn't block access — CA does.

3. What's the difference between Entra registered, joined, and hybrid joined?

Registered = user-owned BYOD; Joined = org-owned cloud-only; Hybrid = on-prem AD + Entra transition state.

4. How does BitLocker recovery work in Intune?

Keys are escrowed to Entra ID; if a device hits the recovery screen, an admin or user retrieves the 48-digit key from Entra to unlock it.

5. Why go passwordless?

Passwordless methods (Windows Hello, FIDO2) are phishing-resistant — there's no shared secret to steal — and they're more convenient.

Scenario-Based Questions

1. Only managed laptops should reach SharePoint. Design it.

A compliance policy defines healthy (encryption, OS, Defender); a Conditional Access policy requires a compliant device for SharePoint, blocking unmanaged machines automatically.

2. Allow BYOD email without managing the whole phone.

Entra-register the device and apply app protection (MAM); CA requires a registered device with an app protection policy, containing corporate data at the app level.

3. Executives are being phished. Harden them.

Issue FIDO2 keys and require phishing-resistant authentication strength in CA for their accounts and admin portals.

4. Tightening a compliance rule would lock out hundreds at once.

Add a grace period so devices have time to remediate before being marked noncompliant.

Troubleshooting Questions

1. A user is blocked and you suspect Conditional Access. Where do you look?

Entra sign-in logs, Conditional Access tab, for that sign-in — it shows which policy and control blocked access.

2. A device is noncompliant but looks healthy. Next step?

Open its per-setting compliance detail in Intune to find the exact failing rule, then remediate that setting.

3. BitLocker recovery is needed but no key is in Entra. Cause?

Encryption occurred before escrow was configured or escrow failed; back up the existing key to Entra and enforce escrow.

Practical Administrator Questions

1. How do you target a policy to one department automatically?

Use a dynamic group with a rule like `user.department -eq "Finance"`.

2. How do you stop MFA-fatigue attacks?

Enable number matching and additional context in Microsoft Authenticator.

3. How do you reduce password-reset tickets?

Enable self-service password reset (SSPR) with registered methods (and writeback for hybrid).

Production Environment Questions

1. How do you safely deploy a strict CA policy org-wide?

Exclude break-glass accounts, run it in report-only, review sign-in insights, then enable On — keeping break-glass available for recovery.

2. How do you prove fleet encryption to an auditor?

Export the encryption report (or pull via Graph) showing BitLocker status across all devices.

3. How do you minimize standing admin privilege?

Use Privileged Identity Management (Entra ID P2) for just-in-time, approval-based role activation.

Summary

Expect deep questions on the compliance-to-Conditional-Access pipeline, the three device identity states, BitLocker escrow and recovery, and passwordless/MFA strengthening. Answer with the business goal, the specific feature, and the exact portal path or tool.

10. Section 2 — Troubleshooting Reference

Introduction

A symptom-first reference for the identity and compliance issues you'll meet most. Each entry follows Symptoms → Root Cause → Steps → Resolution.

1. Compliance policy not applying

Symptoms: Device shows "Not evaluated" or no policy.

Root Cause: Not assigned to the device's group, not enrolled, wrong platform, or not synced.

Steps: Verify assignment groups and platform; confirm enrollment; force a sync.

Resolution: Fix assignment/enrollment; status evaluates on next sync.

2. Conditional Access blocking legitimate users

Symptoms: "Access blocked by Conditional Access."

Root Cause: Overly broad policy, noncompliant/unregistered device, or misconfigured condition.

Steps: Check the sign-in logs CA tab to see the exact policy/control; verify device compliance/registration.

Resolution: Adjust scope/exclusions or remediate the device; re-test in report-only.

3. Device unexpectedly noncompliant

Symptoms: Healthy-looking device flagged noncompliant.

Root Cause: A specific rule failing (Defender off, OS below minimum, encryption pending) or grace period expired.

Steps: Open per-setting compliance detail to find the failing rule.

Resolution: Remediate that setting; device re-evaluates to Compliant.

4. BitLocker not encrypting

Symptoms: Devices remain unencrypted despite policy.

Root Cause: No/disabled TPM, unmet prerequisites, unassigned policy, or escrow failing.

Steps: Check TPM (`Get-Tpm`); confirm assignment; review encryption report; verify escrow.

Resolution: Enable TPM, fix assignment/escrow; encryption proceeds.

5. BitLocker recovery key missing in Entra

Symptoms: Recovery prompt but no key in Entra.

Root Cause: Encryption preceded escrow, or escrow failed.

Steps: Check local protectors (`manage-bde -protectors -get C:`); back up the key to Entra.

Resolution: Escrow existing key and enforce escrow going forward.

6. User can't complete MFA

Symptoms: No push, or method unavailable.

Root Cause: Method not registered, notifications off, time skew, or method disabled in policy.

Steps: Confirm a registered method; check the app; verify the method is enabled/scoped.

Resolution: Re-register/enable the method; use SSPR or admin reset for lockouts.

7. Windows Hello won't set up

Symptoms: Can't enroll PIN/biometric.

Root Cause: Missing/disabled TPM, WHfB policy not applied, or improper join.

Steps: Confirm TPM enabled; verify policy assignment; confirm join with `dsregcmd /status`.

Resolution: Enable TPM, ensure policy and join, then re-enroll the gesture.

8. User not getting expected policy/license

Symptoms: Missing app, license, or policy others have.

Root Cause: Not in the targeting group, blank dynamic-rule attribute, or processing delay.

Steps: Check group membership; verify the attribute the rule relies on; allow dynamic evaluation.

Resolution: Correct the attribute or add to the group; assignments follow.

Best Practices

- Start access investigations in the **Entra sign-in logs (CA tab)**.
- Start compliance investigations in the **per-setting** device detail.
- Always check **assignment scope** and **sync state** before assuming a feature is broken.
- Keep **break-glass accounts** and use **report-only** to avoid lockouts.
- Ensure **BitLocker escrow** is required before encryption everywhere.

Summary

Most Section 2 issues trace to assignment/scope gaps, missing sync, unmet prerequisites (TPM), unregistered methods, or Conditional Access misconfiguration. A symptom-first approach plus the sign-in logs CA tab and per-setting compliance detail resolves the majority quickly.

End of Section 2 — Manage Identity and Compliance.
Next: *Section 3 — Manage, Maintain, and Protect Devices* (Intune overview, enrollment, configuration profiles, administrative templates, device restrictions, endpoint security policies, Microsoft Defender for Endpoint, remote actions, device monitoring, Windows Update for Business). Request it whenever you're ready and it will follow the same structure.

MICROSOFT CERTIFICATION

MD-102

Microsoft Endpoint Administrator

Section 3 — Manage, Maintain, and Protect Devices

A Practical, Interview-Focused Enterprise Study Guide

Intune · Enrollment · Configuration Profiles · Administrative Templates · Device Restrictions

Endpoint Security · Defender for Endpoint · Remote Actions · Monitoring · Windows Update

Real-World Administration • Troubleshooting • Hands-On Labs • Interview Preparation

Section 3 Contents

1. Microsoft Intune Overview
2. Device Enrollment
3. Configuration Profiles
4. Administrative Templates
5. Device Restrictions
6. Endpoint Security Policies
7. Microsoft Defender for Endpoint
8. Remote Actions
9. Device Monitoring
10. Windows Update for Business
11. Section 3 — Consolidated Interview Preparation
12. Section 3 — Troubleshooting Reference

How to use this section: This is the operational heart of endpoint administration — enrolling devices, shaping them with policy, protecting them, and keeping them healthy. Each topic keeps the fixed structure (concepts, architecture, exact portal navigation, step-by-step configuration, scenarios, enterprise examples, troubleshooting, best practices, security, PowerShell, interview Q&A, exam notes, hands-on lab, summary, mini quiz). Build every lab in a trial Intune tenant.

1. Microsoft Intune Overview

Introduction

Microsoft Intune is Microsoft's cloud-based **endpoint management** service. It lets you enroll, configure, secure, and monitor devices — Windows, macOS, iOS/iPadOS, Android — and the apps on them, all from one web console with no on-premises servers. Intune is the engine that delivers everything you've read about so far: it pushes compliance policies, configuration, encryption, apps, and updates to devices and reports their state back to Entra ID. It is the single most important product in MD-102.

Why This Topic Is Important

Intune *is* the job. Almost every task an endpoint administrator performs happens in the Intune Admin Center. Understanding what Intune is, how MDM and MAM differ, and how policies flow to devices is the foundation everything else builds on. Expect the exam and interviews to assume deep Intune familiarity.

Key Concepts

- **MDM (Mobile Device Management):** Managing the whole device — configuration, compliance, apps, wipe. Used for corporate-owned devices.
- **MAM (Mobile Application Management):** Managing only the corporate *apps and data*, not the whole device. Used for BYOD via app protection policies.
- **Enrollment:** Bringing a device under Intune management.
- **Policies:** Compliance, configuration, endpoint security, app, and update policies pushed to devices.
- **Assignments:** Targeting policies/apps to Entra user or device groups.
- **Tenant:** Your organization's Intune/Entra instance.
- **Co-management:** Managing a device jointly with Configuration Manager and Intune.
- **RBAC in Intune:** Scoped admin roles (e.g., Help Desk Operator, Policy and Profile Manager) plus scope tags to limit what admins see.

MDM vs MAM — Comparison

Aspect	MDM (Device Management)	MAM (App Management)
Manages	The entire device	Only corporate apps & data
Enrollment	Device enrolled in Intune	No full enrollment required
Best for	Corporate-owned devices	BYOD / personal devices
Example controls	Compliance, configuration, full wipe	App protection, selective wipe of app data
User privacy	Org sees/controls device	Personal data untouched
Wipe scope	Whole device	Only corporate app data

Architecture / Workflow

1. Devices **enroll** into Intune (or apps are protected via MAM).
2. Admin creates **policies** and **assigns** them to Entra groups.
3. The Intune service delivers policy to devices at **check-in/sync**.
4. Devices apply settings and report **status** back.
5. Status feeds **compliance** and **Conditional Access**; admins monitor via reports.

Microsoft Portal Navigation

```
Intune Admin Center (intune.microsoft.com) → Home
```

```
Intune Admin Center → Devices (all devices, enrollment, compliance, configuration)
```

```
Intune Admin Center → Apps (app deployment and protection)
```

```
Intune Admin Center → Endpoint security (security baselines, disk encryption,
antivirus, firewall)
```

```
Intune Admin Center → Tenant administration → Roles (RBAC)
```

Step-by-Step Configuration

A. Set the MDM authority and automatic enrollment

1. Confirm Intune is the MDM authority (default for new tenants).
2. Go to *Devices* → *Enrollment* → *Windows* → *Automatic Enrollment*; set MDM user scope to All.

B. Create a scoped admin role (RBAC)

1. Go to *Tenant administration* → *Roles*.
2. Assign a built-in role (e.g., **Help Desk Operator**) to a help-desk group.
3. Use **scope tags** to limit which objects/devices that role can see.

Real-World Scenario

> **Scenario: One console, mixed fleet**
>
> A company manages Windows laptops, iPhones, and Android tablets. Corporate laptops and tablets are MDM-enrolled for full control; employees' personal phones use MAM app protection so only Outlook/Teams data is governed. The admin runs everything — config, compliance, apps, security — from the single Intune Admin Center.

Enterprise Deployment Example

A 20,000-seat enterprise uses Intune as its unified endpoint platform: Autopilot provisioning, MDM for corporate devices, MAM for BYOD, security baselines for hardening, and Windows Update rings for

patching. RBAC with scope tags lets regional help desks manage only their own devices, while the central team retains global policy control. Co-management bridges legacy Configuration Manager during migration.

Troubleshooting

Policies not reaching a device

Symptoms: Device enrolled but missing expected settings/apps.

Root Cause: Policy not assigned to the device's group, licensing gap, or device hasn't synced.

Steps: Verify assignments; confirm the user/device has an Intune license; force a sync; check the platform matches.

Resolution: Fix assignment/licensing; settings arrive on next check-in.

Admin sees too much or too little

Symptoms: Help-desk admin can edit global policy, or can't see their devices.

Root Cause: Wrong RBAC role or missing/incorrect scope tags.

Steps: Review the assigned role and scope tags; align them to the admin's responsibility.

Resolution: Apply least-privilege role + correct scope tags.

Best Practices

- Use **MDM for corporate devices**, **MAM for BYOD**.
- Apply **RBAC + scope tags** for least-privilege delegated administration.
- Target policies with **dynamic groups** for accuracy.
- Standardize on the Intune Admin Center as the single pane of glass.
- Plan co-management as a bridge, aiming for cloud-native over time.

Security Considerations

Intune is a powerful control plane — securing admin access is critical. Enforce MFA and PIM for Intune admins, use least-privilege roles, and audit changes. MAM lets you protect corporate data on unmanaged devices without over-reaching into personal data, an important privacy and security balance for BYOD.

PowerShell Commands

```
# Connect to Intune via Microsoft Graph
Connect-MgGraph -Scopes
"DeviceManagementManagedDevices.Read.All","DeviceManagementConfiguration.Read.All"

# List managed devices
Get-MgDeviceManagementManagedDevice | Select-Object DeviceName, ManagedDeviceOwnerType,
OperatingSystem

# List configuration policies
Get-MgDeviceManagementDeviceConfiguration | Select-Object DisplayName
```

Interview Questions and Answers

BEGINNER **What is Microsoft Intune?**

A cloud-based endpoint management service for enrolling, configuring, securing, and monitoring devices and apps from one console, with no on-prem servers.

BEGINNER **Difference between MDM and MAM?**

MDM manages the whole device (corporate-owned); MAM manages only corporate apps and data (BYOD), leaving personal data untouched.

INTERMEDIATE **How do policies get from Intune to a device?**

Admins assign policies to Entra groups; the device receives them at check-in/sync, applies them, and reports status back.

INTERMEDIATE **What are scope tags used for?**

To limit which objects/devices an RBAC role can see, enabling delegated administration (e.g., regional help desks see only their devices).

ADVANCED **When would you choose MAM-only over full MDM?**

For BYOD/personal devices where the org must protect corporate data but shouldn't manage or wipe the entire personal device — app protection policies handle this.

MD-102 Exam Notes

Exam Focus

Intune = cloud endpoint management. Know **MDM vs MAM** cold (whole device vs app data; corporate vs BYOD; full wipe vs selective wipe). Policies are **assigned to groups** and applied at **sync**. **RBAC roles + scope tags** enable delegated, least-privilege administration.

Hands-On Lab

Lab 1.1 — Explore Intune

1. Sign in to the Intune Admin Center and tour Devices, Apps, and Endpoint security. 2. Confirm automatic enrollment is set (Devices → Enrollment → Windows). 3. In Tenant administration → Roles, assign Help Desk Operator to a test group. 4. Create a scope tag and apply it to a policy. 5. Run the Graph commands to list managed devices.

Summary

Microsoft Intune is the cloud endpoint management platform at the centre of MD-102. It supports MDM for corporate devices and MAM for BYOD, delivers policies by assigning them to groups, reports status into compliance and Conditional Access, and uses RBAC with scope tags for delegated administration — all from a single console.

Mini Quiz

1. What does Intune manage and from where?
2. Which model manages only corporate app data?
3. How are policies targeted to devices?
4. When do devices receive policy?
5. What limits which devices an admin role can see?

Answers: 1) Devices and apps, from the cloud Intune Admin Center. 2) MAM. 3) By assigning to Entra groups. 4) At check-in/sync. 5) Scope tags.

2. Device Enrollment

Introduction

Device enrollment is the act of bringing a device under Intune management so it can receive policies and apps and report compliance. Different platforms and ownership models use different enrollment methods. Getting enrollment right is essential — an unenrolled device is unmanaged and, with the right Conditional Access, locked out of corporate data.

Why This Topic Is Important

Enrollment is the gateway to management. Choosing the correct method per platform and ownership (corporate vs BYOD) determines how much control you have and how much you respect user privacy. The exam tests enrollment methods across Windows, iOS/iPadOS, and Android in detail.

Key Concepts

- **Automatic enrollment (Windows):** Entra-joined devices auto-enroll via MDM user scope.
- **Bulk enrollment:** Provisioning packages or device enrollment manager accounts for many devices.
- **Apple Automated Device Enrollment (ADE / DEP):** Apple Business Manager + an enrollment profile for zero-touch corporate iOS/macOS.
- **Apple Business Manager (ABM):** Apple portal linking purchased devices and VPP apps to Intune.
- **Android Enterprise modes:** Fully managed, dedicated (kiosk), corporate-owned with work profile (COPE), and personally-owned work profile (BYOD).
- **BYOD enrollment:** User-driven enrollment via Company Portal or MAM-only.
- **Enrollment restrictions:** Rules controlling which platforms/ownership types may enroll and device limits.
- **Company Portal:** The app users use to enroll and access company resources.

Architecture / Workflow

1. Admin configures the right **enrollment method** per platform and sets **enrollment restrictions**.
2. The device enrolls — automatically (Autopilot/Entra join), via the user (Company Portal), or zero-touch (Apple ADE / Android fully managed).
3. Intune issues an MDM certificate/identity and the device becomes managed.
4. Assigned policies and apps flow to the device.
5. The device reports compliance; Conditional Access can require enrollment/compliance.

Enrollment methods by platform

Platform	Corporate (zero/low-touch)	BYOD
Windows	Autopilot + automatic enrollment	Company Portal / Add work account
iOS/iPadOS	Apple ADE via Apple Business Manager	Company Portal (user enrollment)
macOS	Apple ADE	Company Portal
Android	Fully managed / dedicated (kiosk)	Personally-owned work profile

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Enrollment → Windows (Automatic Enrollment, Autopilot, Enrollment restrictions)
```

```
Intune Admin Center → Devices → Enrollment → Apple (Enrollment program tokens, profiles, ABM)
```

```
Intune Admin Center → Devices → Enrollment → Android (Enrollment profiles, managed Google Play)
```

```
Intune Admin Center → Devices → Enrollment → Enrollment device limit / platform restrictions
```

Step-by-Step Configuration

A. Windows automatic enrollment

1. Go to *Devices* → *Enrollment* → *Windows* → *Automatic Enrollment*.
2. Set MDM user scope = All (so Entra-joined devices enroll automatically).

B. Apple ADE (corporate iOS)

1. Set up **Apple Business Manager** and add an MDM server token to Intune.
2. Assign purchased devices to Intune in ABM.
3. In Intune, create an **enrollment profile** and assign it to the devices for zero-touch enrollment.

C. Android Enterprise

1. Connect **managed Google Play** to Intune.
2. Create an enrollment profile for the chosen mode (fully managed, dedicated, or work profile).

D. Enrollment restrictions

1. Go to *Enrollment* → *Enrollment restrictions*.
2. Block unwanted platforms or personal devices; set per-user device limits.

Real-World Scenario

Scenario: Zero-touch iPhones

A field-service company buys 500 iPhones through Apple Business Manager. Each is assigned to Intune in ABM, so when a technician powers one on, it auto-enrolls via Apple ADE, applies restrictions and apps, and is ready — no manual setup, fully supervised.

Scenario: BYOD work profile

Employees enroll personal Android phones using a **work profile**. Corporate apps live in a separate, managed profile; IT can wipe only that profile, leaving personal photos and apps untouched.

Enterprise Deployment Example

A global retailer enrolls Windows via Autopilot, corporate iPads via Apple ADE, shared store Androids as dedicated (kiosk) devices, and staff phones as personally-owned work profiles. Enrollment restrictions block unsupported OS versions and limit personal Windows enrollment, while ABM and managed Google Play feed apps automatically.

Troubleshooting

Windows device won't enroll

Symptoms: Entra-joined but not in Intune.

Root Cause: MDM scope not set, no Intune license, or enrollment restriction blocking.

Steps: Confirm automatic enrollment scope; verify license; check enrollment/platform restrictions; `dsregcmd /status`.

Resolution: Enable scope, assign license, adjust restrictions.

Apple ADE devices not getting the profile

Symptoms: iPhones enroll generically without corporate setup.

Root Cause: Device not assigned to Intune in ABM, expired Apple token, or unassigned enrollment profile.

Steps: Verify ABM device assignment; renew the MDM push/token; confirm profile assignment; sync.

Resolution: Assign device + profile, renew tokens.

Best Practices

- Use **zero-touch** methods (Autopilot, Apple ADE, Android fully managed) for corporate devices.
- Use **work profiles / MAM** for BYOD to protect privacy.
- Set **enrollment restrictions** to block unsupported platforms and limit personal enrollment.
- Keep **Apple tokens** (push certificate, ADE, VPP) from expiring — they break enrollment if lapsed.

• Monitor enrollment failures regularly.

Security Considerations

Enrollment establishes the managed identity that Conditional Access trusts. Restrict who and what can enroll, prefer supervised/fully-managed for corporate hardware (more control), and use work profiles for BYOD to contain corporate data without invading privacy. Expired Apple tokens are a common, avoidable outage — track their renewal dates.

PowerShell Commands

```
# Check Windows MDM enrollment state
dsregcmd /status   # look for MdmUrl / enrollment details

# List enrolled devices and ownership via Graph
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.Read.All"
Get-MgDeviceManagementManagedDevice |
  Select-Object DeviceName, ManagedDeviceOwnerType, EnrolledDateTime, OperatingSystem
```

Interview Questions and Answers

BEGINNER **What is device enrollment?**

Bringing a device under Intune management so it can receive policies and apps and report compliance.

BEGINNER **How do corporate iPhones enroll with zero touch?**

Via Apple Automated Device Enrollment (ADE) using Apple Business Manager and an Intune enrollment profile.

INTERMEDIATE **Name the Android Enterprise enrollment modes.**

Fully managed, dedicated (kiosk), corporate-owned with work profile (COPE), and personally-owned work profile (BYOD).

INTERMEDIATE **What do enrollment restrictions control?**

Which platforms/ownership types may enroll and how many devices per user.

ADVANCED **Apple ADE devices enroll generically — likely causes?**

Device not assigned to Intune in ABM, an expired Apple token, or an unassigned enrollment profile.

MD-102 Exam Notes

> **Exam Focus**
>
> Match method to platform/ownership: Windows = **Autopilot + automatic enrollment**; Apple corporate = **ADE via Apple Business Manager**; Android corporate = **fully managed/dedicated**; BYOD = **work profile / Company Portal / MAM**. Know **enrollment restrictions** and that **Apple tokens expire**.

Hands-On Lab

Lab 2.1 — Configure enrollment

1. Set Windows automatic enrollment scope to All. 2. Review Apple enrollment (note where ABM tokens go). 3. Connect managed Google Play and create an Android work-profile enrollment profile. 4. Create an enrollment restriction blocking a personal platform. 5. Enroll a test Windows device and confirm it appears in Intune.

Summary

Device enrollment brings devices under Intune management, with the right method depending on platform and ownership: Autopilot/automatic enrollment for Windows, Apple ADE for corporate Apple devices, fully managed/work profile for Android, and Company Portal/MAM for BYOD. Enrollment restrictions gate platforms and limits, and Apple tokens must be kept current.

Mini Quiz

1. What does enrollment achieve?
2. Which Apple program enables zero-touch corporate enrollment?
3. Which Android mode suits a shared kiosk?
4. What blocks unsupported platforms from enrolling?
5. What common Apple problem causes enrollment outages?

Answers: 1) Puts a device under Intune management. 2) Apple Automated Device Enrollment (ADE) via Apple Business Manager. 3) Dedicated (kiosk). 4) Enrollment restrictions. 5) Expired Apple tokens (push/ADE/VPP).

3. Configuration Profiles

Introduction

Configuration profiles are how Intune pushes settings to devices — Wi-Fi, VPN, email, certificates, security options, device features, and thousands of granular OS settings. They are the cloud equivalent of Group Policy, but they work over the internet on any enrolled device with no domain controller. The modern profile type, the **Settings Catalog**, exposes a searchable library of every available setting.

Why This Topic Is Important

Configuration profiles are a daily tool — deploying Wi-Fi, certificates, browser settings, and OS controls to the fleet. The exam tests profile types, the Settings Catalog, and how profiles compare to Group Policy. Knowing how to find and assign the right setting is core competence.

Key Concepts

- **Settings Catalog:** The modern, searchable list of all configurable settings — the recommended profile type.
- **Templates:** Pre-built profile types (Wi-Fi, VPN, email, certificates, device restrictions, endpoint protection, etc.).
- **Custom (OMA-URI):** Advanced profiles to configure settings not surfaced elsewhere using OMA-URI paths.
- **Assignment:** Targeting a profile to user or device groups (include/exclude).
- **Conflict resolution:** When two profiles set the same value differently, the conflicting setting is not applied (and is reported as a conflict).
- **Profile vs policy:** Configuration profiles configure settings; compliance policies evaluate health.

Configuration Profiles vs Group Policy — Comparison

Aspect	Configuration Profiles (Intune)	Group Policy (AD)
Delivery	Cloud, over the internet	On-prem, domain network/VPN
Requires	Intune enrollment	Domain join + line of sight to DC
Scope	Any enrolled device anywhere	Domain-joined devices
Setting source	Settings Catalog / templates / OMA-URI	ADMX administrative templates
Best for	Cloud-native, remote devices	Traditional on-prem environments
Conflict handling	Conflicting setting not applied, reported	Precedence (LSDOU / link order)

Architecture / Workflow

1. Admin creates a configuration profile (Settings Catalog, template, or custom).

2. Settings are chosen and the profile is **assigned** to groups.
3. At sync, the device applies the settings via its MDM channel (CSPs).
4. The device reports per-setting success/failure/conflict.
5. Admins review status in the profile's monitoring view.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Configuration → Create → New Policy
```

```
Intune Admin Center → Devices → Configuration → (Settings Catalog / Templates)
```

```
Intune Admin Center → Devices → Configuration → (profile) → Device status / Per-setting status
```

Step-by-Step Configuration

A. Create a Settings Catalog profile (e.g., disable a feature)

1. Go to *Devices* → *Configuration* → *Create* → *New Policy*; Platform = Windows; Profile type = **Settings catalog**.
2. Click *Add settings*, search for the setting, and configure it.
3. Assign to a device group and create.

B. Deploy a Wi-Fi profile (template)

1. Create → New Policy → Templates → **Wi-Fi**.
2. Enter SSID, security type, and credentials/certificate.
3. Assign to devices.

C. Custom OMA-URI (advanced)

1. Create → Templates → **Custom**.
2. Add an OMA-URI setting with its path, data type, and value.
3. Assign and monitor.

Real-World Scenario

Scenario: Push corporate Wi-Fi

New laptops must connect to the corporate Wi-Fi automatically. The admin creates a Wi-Fi configuration profile with the SSID and a certificate, assigns it to all Windows devices, and machines connect seamlessly on first boot — no manual network setup.

Scenario: Replace a GPO

Migrating off Group Policy, the admin recreates a browser-lockdown GPO as a Settings Catalog profile, assigns it to the cloud-native fleet, and decommissions the old GPO.

Enterprise Deployment Example

An enterprise standardizes baseline configuration via Settings Catalog profiles: Wi-Fi, VPN, certificate deployment (SCEP/PKCS), OneDrive known-folder move, and OS feature controls. Profiles are layered by group (baseline for all, extras for specific roles), and per-setting status reports catch conflicts early. Custom OMA-URI handles the rare setting not yet in the catalog.

Troubleshooting

Configuration profile failure

Symptoms: Profile shows Error/Conflict on devices.

Root Cause: Two profiles set the same value differently (conflict), unsupported setting/OS, or syntax error in OMA-URI.

Steps: Open per-setting status to find the failing setting; check for another profile setting the same value; verify OS support and OMA-URI syntax.

Resolution: Remove the conflict or fix the setting; conflicting settings aren't applied until resolved.

Setting not taking effect

Symptoms: Profile reports success but behavior unchanged.

Root Cause: Wrong setting chosen, user vs device context mismatch, or pending sync/restart.

Steps: Confirm the correct setting and assignment context; force sync; restart if required.

Resolution: Correct the setting/assignment; re-evaluate.

Best Practices

- Prefer the **Settings Catalog** for new profiles — searchable and comprehensive.
- Avoid **conflicts** by not setting the same value in multiple profiles.
- Use clear names and a layered approach (baseline + role-specific).
- Pilot profiles before broad assignment; review per-setting status.
- Reserve **custom OMA-URI** for settings not in the catalog/templates.

Security Considerations

Configuration profiles deliver security-relevant settings (certificates, VPN, encryption, feature lockdowns). Protect certificate profiles and credentials, and ensure security baselines aren't accidentally overridden by a conflicting custom profile. Treat profile assignment changes as change-controlled, since a bad profile can weaken posture fleet-wide.

PowerShell Commands

```
# List configuration profiles via Graph
Connect-MgGraph -Scopes "DeviceManagementConfiguration.Read.All"
Get-MgDeviceManagementDeviceConfiguration | Select-Object DisplayName, Id

# Settings Catalog policies are under configuration policies
Get-MgDeviceManagementConfigurationPolicy | Select-Object Name, Id
```

Interview Questions and Answers

BEGINNER **What is a configuration profile?**

An Intune object that pushes settings (Wi-Fi, VPN, certificates, OS options) to enrolled devices — the cloud equivalent of Group Policy.

BEGINNER **What is the Settings Catalog?**

The modern, searchable library of all configurable settings — the recommended profile type.

INTERMEDIATE **How do configuration profiles differ from Group Policy?**

Profiles deliver settings from the cloud to any enrolled device over the internet (no domain controller); Group Policy needs domain join and network line of sight to a DC.

INTERMEDIATE **What happens when two profiles set the same value differently?**

It's a conflict — the conflicting setting is not applied and is reported, so you must resolve it.

ADVANCED **When do you use a custom OMA-URI profile?**

When a needed setting isn't available in the Settings Catalog or templates; you configure it via its OMA-URI path, data type, and value.

MD-102 Exam Notes

Exam Focus

Know profile types: **Settings Catalog** (modern, searchable), **Templates** (Wi-Fi, VPN, certs, restrictions), **Custom OMA-URI** (advanced). Profiles = the cloud Group Policy equivalent but need **enrollment**, not domain join. Conflicting settings are **not applied** and reported. Check **per-setting status** to troubleshoot.

Hands-On Lab

Lab 3.1 — Build configuration profiles

1. Create a Settings Catalog profile that disables a Windows feature; assign it. 2. Create a Wi-Fi template profile with an SSID. 3. (Optional) Create a custom OMA-URI profile. 4. Assign to a test group, sync a device, and review per-setting status. 5. Deliberately create a conflict with a second profile and observe the conflict report.

Summary

Configuration profiles push settings to enrolled devices from the cloud — the modern alternative to Group Policy. The Settings Catalog is the recommended, searchable type; templates cover common needs like Wi-Fi and certificates; custom OMA-URI handles edge cases. Conflicts cause settings to be skipped and reported, and per-setting status is the troubleshooting tool.

Mini Quiz

1. What is the modern, searchable profile type?
2. What's the on-prem equivalent of configuration profiles?
3. What do profiles require that Group Policy doesn't?
4. What happens to a conflicting setting?
5. When do you use custom OMA-URI?

Answers: 1) Settings Catalog. 2) Group Policy. 3) Intune enrollment (not domain join). 4) It isn't applied and is reported as a conflict. 5) For settings not available in the catalog/templates.

4. Administrative Templates

Introduction

Administrative Templates (ADMX-backed policies) in Intune bring the familiar Group Policy experience to the cloud. If you've used the Group Policy editor with its tree of Windows and Office settings, Administrative Templates feel identical — but they're delivered by Intune to enrolled devices. They're ideal for admins migrating from Group Policy who want recognizable ADMX settings without learning new locations.

Why This Topic Is Important

Many settings administrators know from Group Policy (Office, Edge, Windows components) live here in a familiar layout. For organizations moving from on-prem GPO to Intune, Administrative Templates ease the transition and are a tested exam concept distinct from the Settings Catalog.

Key Concepts

- **ADMX-backed settings:** The same administrative template definitions used by Group Policy, surfaced in Intune.
- **Familiar tree layout:** Settings grouped like the GPO editor (Computer/User, Windows Components, Office, Edge).
- **Enabled / Disabled / Not configured:** The classic GPO setting states.
- **Imported custom ADMX:** You can import third-party ADMX files for apps not built in.
- **Relationship to Settings Catalog:** Settings Catalog now includes ADMX settings too; Administrative Templates remain a friendly curated view.

Architecture / Workflow

1. Admin creates an Administrative Templates profile.
2. Settings are toggled Enabled/Disabled in the familiar tree.
3. The profile is assigned to groups and delivered via MDM.
4. Devices apply the ADMX-backed settings and report status.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Configuration → Create → Templates → Administrative Templates
```

```
Intune Admin Center → Devices → Configuration → Import ADMX (custom templates)
```

Step-by-Step Configuration

1. Go to *Devices* → *Configuration* → *Create* → *Templates* → *Administrative Templates*.
2. Browse or search the tree (e.g., Microsoft Edge, Office, Windows Components).
3. Set settings to Enabled/Disabled and configure their values.

4. Assign to a device or user group; create.
5. (Optional) Import a custom ADMX for a third-party app.

Real-World Scenario

Scenario: Lock down Edge

The security team wants to disable the Edge password manager and force a homepage. The admin opens an Administrative Templates profile, finds the Microsoft Edge settings in the familiar tree, sets them Enabled with the desired values, and assigns to all devices — exactly like a GPO, but cloud-delivered.

Enterprise Deployment Example

An enterprise migrating from Group Policy recreates its Office and Edge GPO settings as Administrative Templates profiles, imports a vendor's custom ADMX for a line-of-business app, and assigns them to cloud-native devices — preserving familiar configuration while retiring on-prem GPO.

Troubleshooting

Administrative Template setting not applying

Symptoms: Setting shows configured but no effect, or conflict.

Root Cause: Conflict with another profile/Settings Catalog, wrong context (user vs device), or app not installed.

Steps: Check per-setting status for conflicts; confirm context and that the target app is present; force sync.

Resolution: Resolve conflict / correct context; re-evaluate.

Best Practices

- Use Administrative Templates for **familiar GPO-style** Office/Edge/Windows settings.
- Avoid duplicating the same setting in both an Administrative Template and the Settings Catalog (conflict risk).
- Import **custom ADMX** for third-party apps when needed.
- Document which settings moved from GPO to Intune during migration.

Security Considerations

Administrative Templates often carry security hardening (disabling risky features, enforcing browser policy). Ensure these aren't overridden by conflicting profiles, and treat them as part of your security baseline. Validate imported third-party ADMX from trusted sources only.

PowerShell Commands

```
# Administrative Templates appear among configuration policies via Graph
Connect-MgGraph -Scopes "DeviceManagementConfiguration.Read.All"
Get-MgDeviceManagementGroupPolicyConfiguration | Select-Object DisplayName, Id
```

Interview Questions and Answers

BEGINNER **What are Administrative Templates in Intune?**

ADMX-backed policies that bring the familiar Group Policy settings experience to Intune for enrolled devices.

INTERMEDIATE **Why might a GPO-experienced admin prefer them?**

They use the same ADMX settings and a familiar tree layout (Office, Edge, Windows Components), easing migration from on-prem Group Policy.

ADVANCED **How do Administrative Templates relate to the Settings Catalog?**

Both can deliver ADMX-backed settings; Administrative Templates are a curated, familiar view, while the Settings Catalog is the broader searchable library — avoid configuring the same setting in both to prevent conflicts.

MD-102 Exam Notes

Exam Focus

Administrative Templates = **ADMX-backed**, GPO-like experience in Intune (Enabled/Disabled/Not configured). Great for **Office/Edge/Windows** settings and **GPO migration**. You can **import custom ADMX**. Don't duplicate the same setting in the Settings Catalog (conflict).

Hands-On Lab

Lab 4.1 — Administrative Templates

1. Create an Administrative Templates profile. 2. Configure a Microsoft Edge setting (e.g., disable password manager). 3. Configure an Office setting. 4. Assign to a test group and sync a device. 5. (Optional) Import a custom ADMX and configure one of its settings.

Summary

Administrative Templates deliver ADMX-backed, Group-Policy-style settings through Intune, offering a familiar tree of Office, Edge, and Windows settings ideal for GPO migration. They support custom ADMX import and overlap with the Settings Catalog, so avoid configuring the same setting in both to prevent conflicts.

Mini Quiz

1. What technology backs Administrative Templates?
2. What three states can a setting have?

3. Which admins find them most familiar?
4. Can you add third-party app settings?
5. What should you avoid to prevent conflicts?

Answers: 1) ADMX. 2) Enabled, Disabled, Not configured. 3) Those experienced with Group Policy. 4) Yes, by importing custom ADMX. 5) Configuring the same setting in both Administrative Templates and the Settings Catalog.

5. Device Restrictions

Introduction

Device restriction profiles control what users can and can't do on a managed device — whether the camera works, if the app store is allowed, password rules, cloud backup, copy-paste between work and personal apps, and dozens of other features. They let you enforce corporate policy and reduce risk by turning capabilities on or off across the fleet.

Why This Topic Is Important

Restrictions are how you enforce acceptable-use and security policy on devices — blocking risky features, requiring strong passwords, and locking down kiosks. They're widely used and tested, and they're a frequent interview topic ("how would you stop users from doing X?").

Key Concepts

- **Device restriction profile:** A configuration profile type focused on enabling/disabling device features and setting password rules.
- **Password/passcode rules:** Minimum length, complexity, expiration, idle lock.
- **Feature controls:** Camera, Bluetooth, app store, screenshot, cloud backup, removable storage, etc.
- **Data protection controls:** Restrict copy/paste, save-as, and data flow between managed and unmanaged apps (often via app protection too).
- **Kiosk mode:** Locking a device to a single app or limited set (a specialized restriction/configuration).
- **Platform-specific:** Available restrictions differ across Windows, iOS, and Android.

Architecture / Workflow

1. Admin creates a device restrictions profile for the platform.
2. Features and password rules are configured.
3. The profile is assigned to groups and delivered via MDM.
4. Devices enforce the restrictions and report status.
5. Compliance/Conditional Access can require the resulting posture.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Configuration → Create → Templates → Device restrictions
```

```
Intune Admin Center → Devices → Configuration → Templates → Kiosk (for single/multi-app lockdown)
```

Step-by-Step Configuration

A. Enforce password rules and disable a feature

1. Create → Templates → **Device restrictions**; choose the platform.
2. Set password rules (e.g., minimum length 8, require complexity, idle lock 5 minutes).
3. Disable a feature (e.g., block the camera or app store) as required.
4. Assign to a device group.

B. Configure a kiosk

1. Create → Templates → **Kiosk**.
2. Choose single-app (one app full screen) or multi-app.
3. Specify the allowed app(s) and assign to the kiosk device group.

Real-World Scenario

Scenario: Secure-facility cameras

A defense contractor must disable cameras on all corporate phones inside secure areas. A device restriction profile disables the camera fleet-wide, instantly enforcing the policy with no user action.

Scenario: Reception kiosk

A lobby tablet should only run the visitor check-in app. A single-app kiosk profile locks the device to that app, hiding everything else.

Enterprise Deployment Example

A healthcare provider applies tiered restrictions: clinical tablets block the app store, removable storage, and screenshots and require a strong passcode with short idle lock; shared waiting-room devices run as single-app kiosks; staff phones enforce passcode rules while leaving productivity features enabled. Restrictions are layered with app protection policies to stop data leaving managed apps.

Troubleshooting

Restriction not enforced

Symptoms: Users can still use a feature you blocked.

Root Cause: Profile not assigned, platform doesn't support that restriction, conflict with another profile, or pending sync.

Steps: Confirm assignment and platform support; check per-setting status for conflicts; force sync.

Resolution: Fix assignment/conflict; restriction applies on next sync.

Users locked out by overly strict rules

Symptoms: Complaints about password complexity or features they need.

Root Cause: Restrictions too aggressive for the user group.

Steps: Review the profile against business needs; scope stricter rules to only the groups that require them.

Resolution: Tier restrictions per group rather than one-size-fits-all.

Best Practices

- **Tier** restrictions by role/device type rather than applying the strictest rules everywhere.
- Balance security with usability to avoid help-desk load and shadow IT.
- Combine restrictions with **app protection** for data-leak prevention.
- Use **kiosk** profiles for shared/single-purpose devices.
- Pilot before broad rollout; watch per-setting status.

Security Considerations

Restrictions reduce attack surface (blocking removable storage, untrusted app installs) and enforce baseline hygiene (passcodes, idle lock). Over-restriction can drive users to insecure workarounds, so aim for the minimum that meets the risk. Pair with app protection to control data movement, especially on BYOD.

PowerShell Commands

```
# Device restriction profiles appear among configuration profiles via Graph
Connect-MgGraph -Scopes "DeviceManagementConfiguration.Read.All"
Get-MgDeviceManagementDeviceConfiguration |
  Where-Object { $_.AdditionalProperties['@odata.type'] -like
'*GeneralDeviceConfiguration*' } |
  Select-Object DisplayName
```

Interview Questions and Answers

BEGINNER **What is a device restriction profile?**

A configuration profile that enables/disables device features and sets password rules to enforce corporate policy.

BEGINNER **Give an example of a restriction.**

Blocking the camera, disabling the app store, requiring a minimum passcode length, or enforcing idle lock.

INTERMEDIATE **How would you lock a tablet to a single app?**

Use a single-app kiosk profile specifying the allowed app.

INTERMEDIATE **Why tier restrictions by group?**

Different roles need different controls; one-size-fits-all either under-protects sensitive devices or frustrates users and breeds workarounds.

ADVANCED **A blocked feature still works — what do you check?**

Assignment, platform support for that restriction, conflicts with another profile (per-setting status), and whether the device has synced.

MD-102 Exam Notes

Exam Focus

Device restrictions = a **template** configuration profile for **feature toggles + password rules**. Available settings are **platform-specific**. **Kiosk** profiles lock devices to one or more apps. Conflicts and assignment scope are the usual reasons a restriction doesn't apply.

Hands-On Lab

Lab 5.1 — Apply restrictions

1. Create a device restrictions profile enforcing a minimum passcode and disabling the camera. 2. Assign to a test group and sync. 3. Verify the camera is blocked. 4. Create a single-app kiosk profile for a chosen app. 5. Review per-setting status for any conflicts.

Summary

Device restriction profiles enforce policy by toggling device features and setting password rules, with platform-specific options and kiosk modes for single-purpose devices. Tier them by role to balance security and usability, combine them with app protection for data control, and use per-setting status to resolve conflicts.

Mini Quiz

1. What two main things do device restrictions control?
2. What profile locks a device to one app?
3. Why are available restrictions different across platforms?
4. What complements restrictions for data-leak prevention?
5. Name a common reason a restriction doesn't apply.

Answers: 1) Device features and password rules. 2) A single-app kiosk profile. 3) Each OS exposes different manageable settings. 4) App protection policies. 5) Missing assignment, lack of platform support, or a profile conflict.

6. Endpoint Security Policies

Introduction

The **Endpoint security** node in Intune is a focused workspace for protecting devices. It groups security-specific policies — **security baselines**, antivirus, disk encryption (BitLocker), firewall, endpoint detection and response (EDR), attack surface reduction (ASR), and account protection — so security teams can harden the fleet without wading through general configuration. Think of it as the security control center of Intune.

Why This Topic Is Important

Hardening endpoints is a primary defense against malware and attacks. Endpoint security policies and baselines let you apply Microsoft's recommended security configurations quickly and consistently. This is a high-priority exam area and central to a real security posture.

Key Concepts

- **Security baselines:** Pre-configured sets of Microsoft-recommended security settings (e.g., Windows, Edge, Defender) you can deploy as a hardened starting point.
- **Antivirus policy:** Configures Microsoft Defender Antivirus (real-time protection, scans, exclusions).
- **Disk encryption policy:** BitLocker configuration (covered in Section 2).
- **Firewall policy:** Windows Defender Firewall rules and profiles.
- **Endpoint detection and response (EDR):** Onboards devices to Microsoft Defender for Endpoint.
- **Attack surface reduction (ASR):** Rules that block common attack techniques (e.g., Office macro abuse).
- **Account protection:** Windows Hello and credential protection settings.
- **Baseline versioning:** Baselines update over time; you can review and move to newer versions.

Architecture / Workflow

1. Admin opens **Endpoint security** and picks a policy type or a baseline.
2. Settings are configured (or accepted from the baseline) and **assigned** to groups.
3. Devices apply the security settings via MDM.
4. Devices report status; EDR onboarding connects them to Defender for Endpoint.
5. Compliance/Conditional Access can require the resulting security posture.

Microsoft Portal Navigation

```
Intune Admin Center → Endpoint security → Security baselines
```

```
Intune Admin Center → Endpoint security → Antivirus
```

```
Intune Admin Center → Endpoint security → Disk encryption
```

```
Intune Admin Center → Endpoint security → Firewall
```

```
Intune Admin Center → Endpoint security → Endpoint detection and response
```

```
Intune Admin Center → Endpoint security → Attack surface reduction
```

```
Intune Admin Center → Endpoint security → Account protection
```

Step-by-Step Configuration

A. Deploy a security baseline

1. Go to *Endpoint security* → *Security baselines* → *Security Baseline for Windows* → *Create profile*.
2. Review the recommended settings (adjust only where you have a reason).
3. Assign to a pilot group first, then broadly.

B. Configure antivirus

1. Go to *Endpoint security* → *Antivirus* → *Create policy*; platform Windows.
2. Enable real-time protection, cloud-delivered protection, scheduled scans; set any exclusions carefully.
3. Assign to devices.

C. Enable ASR rules

1. Go to *Endpoint security* → *Attack surface reduction* → *Create policy*.
2. Set rules (e.g., block Office apps from creating child processes) to Block or Audit.
3. Pilot in Audit, then move to Block.

D. Onboard EDR

1. Connect Intune to Microsoft Defender for Endpoint (see Topic 7).
2. Create an *Endpoint detection and response* policy to onboard devices.

Real-World Scenario

Scenario: Fast hardening

A new security lead needs to harden 2,000 laptops quickly. Instead of hand-picking hundreds of settings, they deploy the **Windows security baseline** to a pilot group, validate, then roll it out fleet-wide — applying Microsoft's recommended posture in days.

Scenario: Stop macro malware

After a phishing wave using malicious Office macros, the admin enables ASR rules to block Office apps from spawning child processes and from creating executable content — neutralizing the technique across the fleet.

Enterprise Deployment Example

An enterprise builds its security posture entirely in the Endpoint security node: Windows + Edge + Defender baselines as the foundation, an antivirus policy with cloud protection, firewall policies per network profile, ASR rules in Block mode, EDR onboarding to Defender for Endpoint, and account protection enforcing Windows Hello. Baselines are reviewed each version and rolled out via pilot rings.

Troubleshooting

Baseline causing app or login issues

Symptoms: After a baseline, an app breaks or users can't sign in.

Root Cause: A hardened setting is too strict for the environment.

Steps: Identify the offending setting via per-setting status; adjust that specific setting; always pilot baselines first.

Resolution: Tune the baseline; re-deploy to the pilot before broad rollout.

ASR rule blocking a legitimate app

Symptoms: A business app is blocked after enabling ASR.

Root Cause: The app's behavior matches an ASR rule.

Steps: Review ASR reports to find the triggering rule; add a tightly scoped exclusion or run that rule in Audit.

Resolution: Add a justified exclusion; keep other rules in Block.

Best Practices

- Start hardening with **security baselines**, then tune.
- Pilot ASR rules in **Audit** before **Block**.
- Minimize antivirus **exclusions** — each is a potential gap.
- Keep baselines on **current versions**; review changes when Microsoft updates them.
- Layer: baseline + antivirus + firewall + ASR + EDR + account protection.

Security Considerations

The Endpoint security node is your hardening toolkit; misuse can both over- and under-protect. Excessive exclusions or disabled protections create real risk, while overly strict baselines break apps — so pilot and tune. ASR and EDR are high-value: ASR blocks common attack techniques and EDR provides detection/response. Treat all changes as security-significant and change-controlled.

PowerShell Commands

```
# List security baselines and intents via Graph
Connect-MgGraph -Scopes "DeviceManagementConfiguration.Read.All"
Get-MgDeviceManagementIntent | Select-Object DisplayName, TemplateId

# Check Defender AV status on a device
Get-MpComputerStatus | Select-Object AMRunningMode, RealTimeProtectionEnabled,
AntivirusEnabled
# View configured ASR rules
Get-MpPreference | Select-Object -ExpandProperty AttackSurfaceReductionRules_Ids
```

Interview Questions and Answers

BEGINNER **What is the Endpoint security node?**

A focused area of Intune for security policies — baselines, antivirus, encryption, firewall, EDR, ASR, and account protection.

BEGINNER **What is a security baseline?**

A pre-configured set of Microsoft-recommended security settings you deploy as a hardened starting point.

INTERMEDIATE **What are ASR rules?**

Attack surface reduction rules that block common attack techniques, like Office apps spawning child processes; pilot in Audit before Block.

INTERMEDIATE **Why minimize antivirus exclusions?**

Each exclusion is an unscanned area attackers can exploit, so only add well-justified ones.

ADVANCED **A new baseline breaks an app — how do you handle it?**

Use per-setting status to find the offending hardened setting, adjust just that setting, and always pilot baselines before broad rollout.

MD-102 Exam Notes

Exam Focus

Endpoint security policy types: **security baselines**, **antivirus**, **disk encryption**, **firewall**, **EDR**, **ASR**, **account protection**. Baselines = Microsoft-recommended settings. ASR: **Audit then Block**. EDR onboards to **Defender for Endpoint**. Pilot baselines; minimize exclusions.

Hands-On Lab

Lab 6.1 — Harden with Endpoint security

1. Deploy the Windows security baseline to a pilot group. 2. Create an antivirus policy enabling real-time and cloud protection. 3. Create an ASR policy in Audit mode. 4. Review ASR/baseline per-setting status. 5. (If Defender for Endpoint is connected) create an EDR onboarding policy.

Summary

Endpoint security policies are Intune's hardening toolkit: baselines apply Microsoft-recommended settings, while antivirus, firewall, disk encryption, ASR, EDR, and account protection layer defense in depth. Pilot baselines and ASR (Audit before Block), minimize exclusions, and keep baselines current to maintain a strong, stable posture.

Mini Quiz

1. What does a security baseline provide?
2. What do ASR rules do?
3. In what mode should you pilot ASR rules?
4. What does an EDR policy onboard devices to?
5. Why limit antivirus exclusions?

Answers: 1) A pre-configured set of Microsoft-recommended security settings. 2) Block common attack techniques. 3) Audit mode (then Block). 4) Microsoft Defender for Endpoint. 5) Each exclusion is a potential security gap.

7. Microsoft Defender for Endpoint

Introduction

Microsoft Defender for Endpoint (MDE) is Microsoft's enterprise **endpoint detection and response (EDR)** platform. While Defender Antivirus prevents and removes known malware, Defender for Endpoint goes further: it continuously monitors device behavior, detects advanced attacks, calculates a **risk score** per device, surfaces a **Microsoft Secure Score** and vulnerability data, and enables investigation and automated response. Integrated with Intune, its risk signals can drive compliance and Conditional Access.

Why This Topic Is Important

Modern attacks evade simple antivirus. EDR provides the detection, investigation, and response capability enterprises need, and its integration with Intune (risk-based compliance) is a powerful, exam-relevant pattern. Understanding the AV-vs-EDR distinction and the onboarding/integration flow is essential.

Key Concepts

- **EDR:** Endpoint Detection and Response — behavioral monitoring, detection, investigation, and response.
- **Onboarding:** Connecting devices to the MDE service (via an Intune EDR policy and the service connection).
- **Device risk score:** MDE's assessment of a device's risk (e.g., Low/Medium/High) based on detections and vulnerabilities.
- **Threat & Vulnerability Management (TVM):** Identifies vulnerabilities and misconfigurations with remediation guidance.
- **Automated investigation & response (AIR):** Automatically investigates alerts and can remediate.
- **Microsoft Secure Score for Devices:** A measure of device security posture.
- **Intune connector:** The integration that lets MDE risk feed Intune compliance.

Defender Antivirus vs Defender for Endpoint — Comparison

Aspect	Defender Antivirus	Defender for Endpoint (EDR)
Primary job	Prevent/remove known malware	Detect, investigate, respond to advanced threats
Approach	Signature + heuristic prevention	Behavioral monitoring + analytics
Visibility	Per-device protection	Fleet-wide detection, risk scores, vuln mgmt
Response	Quarantine/clean	Investigate, isolate device, automated remediation
Licensing	Built into Windows	Requires MDE licensing (e.g., E5)
Intune integration	Configured via antivirus policy	Risk score feeds compliance/Conditional Access

Architecture / Workflow

1. Connect Intune and Microsoft Defender for Endpoint (the service-to-service connector).
2. Create an Intune **EDR (onboarding)** policy to onboard devices.
3. Onboarded devices stream telemetry to MDE, which detects threats and computes a **risk score**.
4. A **compliance policy** sets a maximum allowed machine risk (e.g., "Medium or lower").
5. **Conditional Access** blocks devices that exceed the risk threshold until remediated.
6. Analysts investigate alerts and use AIR to respond.

Microsoft Portal Navigation

```
Intune Admin Center → Endpoint security → Microsoft Defender for Endpoint (connector settings)
```

```
Intune Admin Center → Endpoint security → Endpoint detection and response → Create policy (onboarding)
```

```
Microsoft Defender portal (security.microsoft.com) → Assets → Devices
```

```
Microsoft Defender portal → Vulnerability management / Secure Score
```

Step-by-Step Configuration

A. Connect Intune to MDE

1. Go to *Endpoint security* → *Microsoft Defender for Endpoint*.
2. Open the connection to the Defender portal and turn on the Intune connection.
3. Enable "Allow Microsoft Defender for Endpoint to enforce Endpoint Security Configurations" and compliance evaluation.

B. Onboard devices (EDR policy)

1. Go to *Endpoint security* → *Endpoint detection and response* → *Create policy*.
2. The onboarding configuration is pulled from the connector; assign to device groups.

C. Use risk in compliance + Conditional Access

1. In the Windows compliance policy, set **Machine risk score** = Medium (or your threshold).
2. Ensure a Conditional Access policy requires compliant devices, so high-risk devices are blocked.

Real-World Scenario

> **Scenario: Risk-based lockout**
>
> A laptop is infected and MDE raises its risk to **High**. Because compliance requires "Medium or lower" and Conditional Access requires compliance, the device is automatically blocked from corporate apps until the threat is remediated and the risk drops — containing the incident without manual intervention.

Scenario: Vulnerability remediation

TVM flags that many devices run an outdated, vulnerable app version. The admin uses the remediation guidance and an Intune app update to patch the fleet, raising Secure Score.

Enterprise Deployment Example

An E5 enterprise onboards all Windows endpoints to MDE via Intune, ties machine risk into compliance and Conditional Access, runs ASR rules and TVM for proactive hardening, and uses automated investigation and response to triage alerts at scale. The SOC works incidents in the Defender portal while Intune enforces device-level remediation gates.

Troubleshooting

Devices not onboarding to MDE

Symptoms: Devices don't appear in the Defender portal.

Root Cause: Connector not enabled, EDR policy unassigned, licensing missing, or connectivity blocked.

Steps: Confirm the Intune–MDE connection is on; verify EDR policy assignment and MDE licensing; check device connectivity to Defender endpoints.

Resolution: Enable connector, assign onboarding policy, license, and allow required URLs.

Risk score not affecting access

Symptoms: High-risk device still has access.

Root Cause: Compliance policy doesn't include machine risk, or Conditional Access doesn't require compliance.

Steps: Add machine risk to the compliance policy; confirm CA requires a compliant device; verify the connector passes risk to Intune.

Resolution: Wire risk → compliance → Conditional Access end to end.

Best Practices

- Integrate **MDE risk with Intune compliance** and Conditional Access for automated containment.
- Run **TVM** and act on remediation guidance to raise Secure Score.
- Enable **automated investigation and response** to reduce analyst load.
- Combine MDE with **ASR** and security baselines for layered defense.
- Monitor onboarding coverage — unonboarded devices are blind spots.

Security Considerations

EDR is a cornerstone of modern defense, providing visibility and response that antivirus alone cannot. The highest-value pattern is risk-based Conditional Access: a compromised device loses access automatically.

Ensure full onboarding coverage, protect the Defender portal with strong admin controls, and treat risk thresholds as a deliberate security decision.

PowerShell Commands

```
# Verify MDE onboarding/sense service on a device
Get-Service -Name Sense | Select-Object Name, Status

# Check onboarding state in registry (high level)
Get-ItemProperty "HKLM:\SOFTWARE\Microsoft\Windows Advanced Threat Protection\Status" -
ErrorAction SilentlyContinue

# Defender AV status (works alongside EDR)
Get-MpComputerStatus | Select-Object RealTimeProtectionEnabled, AMRunningMode
```

Interview Questions and Answers

BEGINNER **What is Microsoft Defender for Endpoint?**

Microsoft's enterprise EDR platform that detects, investigates, and responds to advanced threats and provides device risk scores and vulnerability management.

BEGINNER **How does it differ from Defender Antivirus?**

Antivirus prevents/removes known malware; Defender for Endpoint adds behavioral detection, fleet-wide visibility, risk scoring, and response.

INTERMEDIATE **How does MDE integrate with Intune?**

Through a connector that feeds device risk into Intune compliance; Conditional Access then blocks devices over a risk threshold.

INTERMEDIATE **What is TVM?**

Threat & Vulnerability Management — it identifies vulnerabilities/misconfigurations and provides remediation guidance.

ADVANCED **Design automated containment of a compromised device.**

Onboard devices to MDE, set a machine-risk rule in the compliance policy, and require compliant devices in Conditional Access — so a high-risk device is blocked automatically until remediated.

MD-102 Exam Notes

Exam Focus

MDE = **EDR** (detect/investigate/respond) vs Defender AV = prevention. Key flow: **connector → EDR onboarding policy → device risk score → compliance (machine risk) → Conditional Access**. Know **TVM**, **Secure Score**, and **automated investigation & response**. MDE needs appropriate licensing (e.g., E5).

Hands-On Lab

Lab 7.1 — Integrate MDE with Intune

1. In Endpoint security → Microsoft Defender for Endpoint, enable the Intune connection. 2. Create an EDR onboarding policy and assign it. 3. Confirm a test device appears in the Defender portal under Devices. 4. Add a machine-risk rule (e.g., Medium) to the Windows compliance policy. 5. Confirm Conditional Access requires compliant devices.

Summary

Microsoft Defender for Endpoint is the EDR platform that detects and responds to advanced threats, scores device risk, and manages vulnerabilities. Integrated with Intune, its risk feeds compliance and Conditional Access to automatically contain compromised devices. It complements Defender Antivirus (prevention) with detection, investigation, and response.

Mini Quiz

1. What does EDR stand for?
2. What's the main difference between Defender AV and MDE?
3. What MDE signal feeds Intune compliance?
4. What blocks a high-risk device from data?
5. What identifies vulnerabilities with remediation guidance?

Answers: 1) Endpoint Detection and Response. 2) AV prevents known malware; MDE detects/investigates/responds to advanced threats. 3) The device risk score. 4) Conditional Access (via a compliance machine-risk rule). 5) Threat & Vulnerability Management (TVM).

8. Remote Actions

Introduction

Remote actions are commands you send from the Intune console to a managed device without touching it physically — wipe, retire, lock, reset the passcode, restart, sync, locate, and more. They're how administrators respond to lost/stolen devices, prepare hardware for reuse, and resolve issues for remote users.

Why This Topic Is Important

When a device is lost, stolen, or being reassigned, remote actions are the immediate response. Knowing the difference between **Wipe**, **Retire**, and **Fresh Start** — and when to use each — is a classic exam and interview topic with real security consequences.

Key Concepts

- **Wipe:** Factory-resets the device, removing all data (corporate and personal). Option to retain enrollment or remove entirely.
- **Retire:** Removes only corporate data/management, leaving personal data intact — ideal for BYOD offboarding.
- **Fresh Start (Windows):** Reinstalls Windows keeping user data but removing pre-installed apps, returning the device to a clean state.
- **Remote Lock:** Locks the device immediately.
- **Reset Passcode:** Clears/resets the device passcode.
- **Sync:** Forces the device to check in for policy.
- **Locate device / Lost mode:** Finds a device (supervised iOS) or puts it in lost mode.
- **Restart / Collect diagnostics:** Operational helpers.

Wipe vs Retire vs Fresh Start

Action	What it removes	Best for
Wipe	Everything — factory reset (corporate + personal)	Lost/stolen or fully repurposing a corporate device
Retire	Only corporate data & management	BYOD offboarding; keep personal data
Fresh Start	Pre-installed/OEM apps; keeps user data (Windows)	Cleaning up a Windows device while keeping files

Architecture / Workflow

1. Admin selects a device in Intune and chooses a remote action.
2. Intune queues the command; the device executes it at next contact (or immediately if online).
3. The action completes (e.g., data removed, device locked) and status is reported.

4. For wipe/retire, the device object can be removed from management.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → All devices → (select device) → (Wipe / Retire / Fresh Start / Remote lock / Sync / …)
```

```
Intune Admin Center → Devices → (device) → Locate device (supervised iOS)
```

Step-by-Step Configuration

A. Wipe a lost corporate device

1. Go to *Devices* → *All devices* and select the device.
2. Choose **Wipe**; decide whether to retain enrollment.
3. Confirm; the device factory-resets at next contact.

B. Retire a BYOD device at offboarding

1. Select the personal device → **Retire**.
2. Corporate data/management is removed; personal data remains.

C. Other actions

1. Use **Remote lock** for a temporarily misplaced device, **Reset passcode** for lockouts, and **Sync** to force policy refresh.

Real-World Scenario

Scenario: Stolen laptop

A corporate laptop is stolen. The admin issues a **Wipe**; combined with BitLocker, the data is unreadable and the device is reset the moment it next connects. The device is then removed from management and replaced.

Scenario: Employee leaves (BYOD)

A departing employee used their personal phone for work. The admin issues **Retire**, which removes corporate email and app data while leaving the employee's personal photos and apps untouched.

Enterprise Deployment Example

An enterprise builds offboarding and incident runbooks around remote actions: lost/stolen corporate devices are wiped, BYOD leavers are retired, reassigned Windows devices get Fresh Start, and help desk uses remote lock/reset passcode/sync for day-to-day support. Actions are role-delegated via RBAC and audited.

Troubleshooting

Remote action pending / not completing

Symptoms: Wipe or lock stays "Pending."

Root Cause: Device offline; commands execute only when the device next contacts Intune.

Steps: Confirm the action is queued; understand it will run when the device reconnects; for stolen devices, the wipe applies on next online event.

Resolution: Wait for device contact; the action completes then.

Wrong action used (data loss risk)

Symptoms: Personal data removed from a BYOD device.

Root Cause: Wipe used instead of Retire on a personal device.

Steps: Always match action to ownership — Retire for BYOD, Wipe for corporate; confirm before executing.

Resolution: Use Retire for personal devices to avoid removing personal data.

Best Practices

- **Wipe** corporate devices; **Retire** BYOD — match the action to ownership.
- Pair wipe with **BitLocker** so data is unreadable even before the wipe runs.
- Document **offboarding/incident runbooks** so the right action is used under pressure.
- Delegate remote actions via **RBAC** and audit their use.
- Remember actions run at **next device contact** if offline.

Security Considerations

Remote wipe is a key data-protection control for lost/stolen devices, strongest when combined with encryption. Restrict who can issue destructive actions (RBAC) and audit them, since a wrongful wipe causes data loss and a missed wipe leaves data exposed. For BYOD, Retire respects privacy while protecting corporate data.

PowerShell Commands

```
# Issue remote actions via Microsoft Graph
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.PrivilegedOperations.All"

# Wipe a device
Invoke-MgWipeDeviceManagementManagedDevice -ManagedDeviceId <id>
# Retire a device (remove corporate data only)
Invoke-MgRetireDeviceManagementManagedDevice -ManagedDeviceId <id>
# Force a sync
Sync-MgDeviceManagementManagedDevice -ManagedDeviceId <id>
```

Interview Questions and Answers

BEGINNER **What are remote actions?**

Commands sent from Intune to a managed device — wipe, retire, lock, reset passcode, sync, etc. — without physical access.

BEGINNER **Difference between Wipe and Retire?**

Wipe factory-resets the whole device (all data); Retire removes only corporate data/management, keeping personal data.

INTERMEDIATE **What is Fresh Start?**

A Windows action that reinstalls Windows, removing pre-installed/OEM apps while keeping user data, returning the device to a clean state.

INTERMEDIATE **A wipe shows Pending for a stolen device — why?**

The device is offline; the command runs the moment it next connects to the internet.

ADVANCED **Which action for a departing employee's personal phone, and why?**

Retire — it removes corporate data and management while preserving the employee's personal data, respecting privacy.

MD-102 Exam Notes

Exam Focus

Know the trio: **Wipe** (everything, corporate device), **Retire** (corporate data only, BYOD), **Fresh Start** (Windows clean-up keeping user data). Actions run at **next device contact** if offline. Pair wipe with **BitLocker**. Other actions: remote lock, reset passcode, sync, locate.

Hands-On Lab

Lab 8.1 — Practice remote actions

1. In Devices → All devices, select a test device. 2. Issue a **Sync** and confirm it completes. 3. Review the available actions and note Wipe vs Retire vs Fresh Start. 4. (On a disposable test device) issue Retire and observe corporate data removal. 5. Document a lost-device runbook.

Summary

Remote actions let administrators manage devices from the console — wipe corporate devices, retire BYOD, Fresh Start Windows machines, lock, reset passcodes, and sync. The right action depends on ownership and scenario, destructive actions should be RBAC-delegated and audited, and offline devices execute commands at next contact. Pair wipe with encryption for strong data protection.

Mini Quiz

1. Which action factory-resets a device entirely?
2. Which action suits BYOD offboarding?
3. Which Windows action keeps user data but removes OEM apps?

4. When does an action run if the device is offline?
5. What should you pair with wipe for strong protection?

Answers: 1) Wipe. 2) Retire. 3) Fresh Start. 4) At the device's next contact with Intune. 5) BitLocker encryption.

9. Device Monitoring

Introduction

Device monitoring is how you keep a finger on the pulse of the fleet: which devices are healthy, which policies succeeded or failed, what's installed, and how users experience their devices. Intune provides device inventory, policy/app status, and **Endpoint analytics** (startup performance, application reliability, and a proactive remediation feature), giving administrators the visibility to catch problems early.

Why This Topic Is Important

You can't manage what you can't see. Monitoring turns raw device data into actionable insight — spotting failing policies, slow boot times, and crashing apps before users complain. Endpoint analytics and reporting are increasingly emphasized and appear on the exam.

Key Concepts

- **Device inventory:** Hardware/software details per device (model, OS, storage, installed apps).
- **Policy/app status:** Success/error/conflict per assignment.
- **Endpoint analytics:** Insights into startup performance, application reliability, and recommended improvements.
- **Proactive remediations:** Script packages (detection + remediation) that automatically fix common issues.
- **Reports:** Organizational, operational, and historical report categories.
- **Log collection / diagnostics:** Pulling device logs for troubleshooting.
- **Microsoft Graph / Power BI:** Exporting data for custom dashboards.

Architecture / Workflow

1. Devices report inventory, policy status, and analytics telemetry to Intune at check-in.
2. Intune aggregates this into **device views, reports, and Endpoint analytics**.
3. Admins review health, identify failures, and drill into specifics.
4. **Proactive remediations** can auto-detect and fix recurring issues.
5. Data can be exported via **Graph/Power BI** for dashboards.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → All devices → (device) → Hardware / Discovered apps
```

```
Intune Admin Center → Reports (Device compliance, Device management, App install status)
```

```
Intune Admin Center → Reports → Endpoint analytics
```

```
Intune Admin Center → Devices → Monitor (policy/app status, encryption, etc.)
```

```
Intune Admin Center → Devices → (Windows) → Proactive remediations (script packages)
```

Step-by-Step Configuration

A. Enable and review Endpoint analytics

1. Go to *Reports* → *Endpoint analytics*; enable it if prompted.
2. Review the startup performance and application reliability scores.
3. Act on the recommended improvements.

B. Create a proactive remediation

1. Go to *Devices* → *(Windows)* → *Proactive remediations* → *Create script package*.
2. Upload a **detection** script and a **remediation** script.
3. Assign to devices and schedule; review run results.

C. Inspect a device and collect diagnostics

1. Open the device → review Hardware and Discovered apps.
2. Use **Collect diagnostics** to pull logs for troubleshooting.

Real-World Scenario

Scenario: Slow boot investigation

Users complain laptops are slow to start. Endpoint analytics shows a high startup time driven by a particular startup process and a slow disk model. The admin disables the process via policy and prioritizes SSD upgrades for the affected model — boot times drop and the analytics score rises.

Scenario: Auto-fix a recurring issue

A misconfigured service keeps stopping on many devices. The admin deploys a **proactive remediation**: a detection script finds the stopped service and a remediation script restarts and re-enables it automatically, eliminating tickets.

Enterprise Deployment Example

An enterprise operationalizes monitoring: Endpoint analytics tracks startup and app reliability across regions, proactive remediations auto-fix common issues, and Intune/Defender data is exported via Graph into a Power BI executive dashboard. Weekly operational reviews target the worst-performing device models and the most common policy failures.

Troubleshooting

Stale or missing monitoring data

Symptoms: Reports show old data or a device missing.

Root Cause: Device hasn't checked in/synced, or analytics not enabled.

Steps: Confirm the device is online and syncing; ensure Endpoint analytics is enabled; allow time for telemetry.

Resolution: After sync/enablement, data populates.

Proactive remediation not running

Symptoms: Script package shows no results.

Root Cause: Not assigned/scheduled, script errors, or platform/edition limitations.

Steps: Verify assignment and schedule; test scripts locally; review run output for errors.

Resolution: Fix assignment/script; results appear after the next run.

Best Practices

- Enable **Endpoint analytics** and act on its recommendations.
- Use **proactive remediations** to auto-fix recurring problems and cut tickets.
- Review **policy/app status** regularly to catch failures early.
- Export to **Power BI/Graph** for leadership-level visibility.
- Use **Collect diagnostics** for deep device troubleshooting.

Security Considerations

Monitoring is also a security function: unexpected app installs, failing security policies, or devices that stop reporting can signal compromise or misconfiguration. Treat monitoring gaps (non-reporting devices) as risks, restrict access to inventory/diagnostic data, and use proactive remediations to keep security controls in their intended state.

PowerShell Commands

```
# Inventory and last sync via Graph
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.Read.All"
Get-MgDeviceManagementManagedDevice |
  Select-Object DeviceName, Model, OSVersion, LastSyncDateTime, ComplianceState

# Discovered apps on a device
Get-MgDeviceManagementManagedDeviceDetectedApp -ManagedDeviceId <id>
```

Interview Questions and Answers

BEGINNER **What does device monitoring give you?**

Visibility into device health, inventory, policy/app status, and user experience so you can catch and fix problems.

BEGINNER **What is Endpoint analytics?**

An Intune feature reporting startup performance, application reliability, and recommended improvements.

INTERMEDIATE **What are proactive remediations?**

Script packages with a detection and a remediation script that automatically find and fix common device issues.

INTERMEDIATE **Why might monitoring data be stale?**

The device hasn't checked in/synced, or the relevant feature (e.g., analytics) isn't enabled.

ADVANCED **How would you give leadership fleet-wide health visibility?**

Export Intune/analytics data via Microsoft Graph into a scheduled Power BI dashboard surfacing compliance, performance, and reliability trends.

MD-102 Exam Notes

> **Exam Focus**
>
> Know **Endpoint analytics** (startup performance, app reliability) and **proactive remediations** (detection + remediation scripts). Use **Reports** and **Devices → Monitor** for policy/app status. **Collect diagnostics** pulls logs. Stale data usually = no sync. Export via **Graph/Power BI**.

Hands-On Lab

> **Lab 9.1 — Monitor the fleet**
>
> 1. Enable Endpoint analytics and review the startup score. 2. Create a proactive remediation (detection + remediation script) and assign it. 3. Open a device and review Hardware and Discovered apps. 4. Run Collect diagnostics on a test device. 5. Explore the Reports hub and export a report.

Summary

Device monitoring provides the visibility to keep the fleet healthy: inventory, policy/app status, Endpoint analytics for performance and reliability, and proactive remediations to auto-fix issues. Administrators review regularly, drill into failures, collect diagnostics for deep troubleshooting, and export to Power BI/Graph for leadership dashboards.

Mini Quiz

1. What feature reports startup performance and app reliability?
2. What auto-detects and fixes issues with scripts?
3. What usually causes stale monitoring data?
4. What pulls device logs for troubleshooting?

5. What do you use to build leadership dashboards?

Answers: 1) Endpoint analytics. 2) Proactive remediations. 3) The device hasn't synced. 4) Collect diagnostics. 5) Microsoft Graph / Power BI.

10. Windows Update for Business

Introduction

Windows Update for Business (WUfB) manages Windows patching from the cloud through Intune. (Section 1 introduced it within deployment; here we focus on the day-to-day management view from the protect-and-maintain perspective.) You define **update rings**, **feature update profiles**, and **driver/quality update** controls, roll changes out in waves, and monitor compliance — keeping devices secure without breaking the business.

Why This Topic Is Important

Patching is the most effective preventable security control, and managing it safely (phased, monitored, reversible) is core endpoint-admin work. WUfB and **Windows Autopatch** are emphasized on the exam, and "how do you patch safely at scale?" is a guaranteed interview question.

Key Concepts

- **Quality updates:** Monthly security/reliability cumulative updates.
- **Feature updates:** Major OS version upgrades; pinned via feature update profiles.
- **Driver updates:** Manage driver delivery (approve/pause) via driver update policies.
- **Update rings:** Groups with deferrals/deadlines for phased rollout (Pilot → Broad).
- **Deadline & grace period:** When install/restart becomes mandatory.
- **Expedite:** Force a critical patch immediately, bypassing deferrals.
- **Windows Autopatch:** Microsoft-managed automation of rings and update orchestration.
- **WUfB reports:** Update compliance and failure visibility.

Architecture / Workflow

1. Intune sends **update policy** (rings/profiles) to devices — content comes from Microsoft.
2. Each device honors its ring's **deferral/deadline**.
3. Pilot rings update first; broad rings follow once validated.
4. **Expedite** handles emergencies; **feature profiles** control version jumps.
5. **Reports** show status; **Autopatch** can automate the whole flow.

Microsoft Portal Navigation

```
Intune Admin Center → Devices → Windows → Windows updates → Update rings
```

```
Intune Admin Center → Devices → Windows → Windows updates → Feature updates
```

```
Intune Admin Center → Devices → Windows → Windows updates → Quality updates
(Expedite)
```

```
Intune Admin Center → Devices → Windows → Windows updates → Driver updates
```

```
Intune Admin Center → Reports → Windows updates
```

Step-by-Step Configuration

A. Pilot and Broad rings

1. Create "Ring 1 - Pilot" (quality deferral ~2 days) → assign to a small group.
2. Create "Ring 2 - Broad" (quality deferral ~7–9 days) → assign to everyone else.
3. Set deadlines and grace periods for restarts.

B. Pin a feature version

1. Create a **Feature updates** profile targeting the desired version; assign to hold devices there.

C. Expedite a critical patch

1. Create an **Expedite** quality update for the specific patch; assign to affected devices.

D. Manage drivers

1. Create a **Driver updates** policy; choose automatic or manual approval.

Real-World Scenario

Scenario: Contained rollout

A monthly update breaks a driver on the pilot ring. Because the broad ring defers 7 days, IT pauses the rollout, fixes the driver, then resumes — sparing the bulk of the fleet.

Scenario: Emergency patch

A zero-day is announced. The admin uses **Expedite** to push the fix within hours, then returns to normal ring schedules.

Enterprise Deployment Example

A 25,000-seat enterprise adopts **Windows Autopatch**: devices auto-distribute across Test/First/Fast/Broad rings, Microsoft orchestrates quality and feature updates, rollouts halt automatically if failure thresholds are crossed, and WUfB reports provide compliance evidence — dramatically reducing manual patch effort.

Troubleshooting

Updates not installing

Symptoms: Devices remain unpatched past the deadline.

Root Cause: Conflicting ring assignments, active pause, connectivity, or low disk space.

Steps: Resolve conflicting rings; lift pauses; check connectivity/disk; review WUfB reports.

Resolution: Single ring authority, free space, force sync.

Device stuck on an old feature version

Symptoms: Won't advance to a newer Windows version.

Root Cause: A feature update profile is pinning it.

Steps: Check the assigned feature profile's target version; update or unassign it.

Resolution: Set the profile to the newer target.

Best Practices

- Always use **multiple rings** (Pilot + Broad) for phased rollout.
- Set realistic **deadlines/grace periods**.
- Use **feature update profiles** to control version jumps.
- Reserve **Expedite** for emergencies.
- Monitor **WUfB reports**; consider **Autopatch** at scale.

Security Considerations

Timely patching closes known vulnerabilities — the biggest preventable risk. Enforce deadlines so users can't defer indefinitely, and tie update compliance to Conditional Access so dangerously out-of-date devices can be flagged non-compliant and lose access until patched.

PowerShell Commands

```
# Recent update history
Get-HotFix | Sort-Object InstalledOn -Descending | Select-Object -First 10

# Manage updates with PSWindowsUpdate
Install-Module PSWindowsUpdate -Force
Get-WindowsUpdate
Install-WindowsUpdate -AcceptAll -AutoReboot
```

Interview Questions and Answers

BEGINNER **What does WUfB manage?**

Windows quality, feature, and driver updates from the cloud via Intune policy (devices get content from Microsoft).

BEGINNER **What is an update ring?**

A device group with its own deferral/deadline settings for phased rollout.

INTERMEDIATE **How do you keep devices on a specific Windows version?**

Assign a feature update profile pinning that version.

INTERMEDIATE **How do you push a critical patch immediately?**

Use Expedite quality updates, which bypasses normal deferrals.

ADVANCED **What does Windows Autopatch add over manual WUfB rings?**

Microsoft-managed ring distribution and orchestration, automatic rollout halting on failures, and built-in reporting — cutting manual effort at scale.

MD-102 Exam Notes

> **Exam Focus**
>
> WUfB delivers **policy, not content**. Know **quality/feature/driver** updates, **rings** + deferrals/deadlines, **feature profiles** (version pinning), **Expedite** for emergencies, and **Windows Autopatch** automation. Tie update compliance to Conditional Access.

Hands-On Lab

> **Lab 10.1 — Manage updates**
>
> 1. Create Pilot and Broad update rings with staggered deferrals. 2. Create a feature update profile pinning a version. 3. Create an Expedite quality update. 4. Create a driver update policy. 5. Review Reports → Windows updates for compliance.

Summary

Windows Update for Business manages quality, feature, and driver updates from the cloud using rings, deferrals, deadlines, and feature profiles for safe phased rollouts. Expedite handles emergencies, reports track compliance, and Windows Autopatch automates the entire process at scale — with update compliance reinforced through Conditional Access.

Mini Quiz

1. What three update types does WUfB manage?
2. What groups devices for phased rollout?
3. How do you pin a Windows version?
4. What pushes a critical patch immediately?
5. What Microsoft service fully automates rings?

Answers: 1) Quality, feature, and driver updates. 2) Update rings. 3) A feature update profile. 4) Expedite quality updates. 5) Windows Autopatch.

11. Section 3 — Consolidated Interview Preparation

Introduction

This domain is where most hands-on endpoint-admin work lives, so interviewers probe it hardest. Below are the most likely questions grouped as employers ask them. Answer each in two or three sentences: concept, Microsoft feature, exact tool or portal path.

Most Asked Interview Questions

1. What is Intune and what does it do?

A cloud endpoint management service to enroll, configure, secure, and monitor devices and apps from one console, delivering policy by assigning it to Entra groups.

2. MDM vs MAM?

MDM manages the whole device (corporate); MAM manages only corporate app data (BYOD), leaving personal data untouched.

3. Configuration profiles vs Group Policy?

Profiles deliver settings from the cloud to any enrolled device; Group Policy needs domain join and DC line of sight. Settings Catalog is the modern profile type.

4. Defender Antivirus vs Defender for Endpoint?

AV prevents/removes known malware; MDE adds behavioral EDR — detection, risk scoring, investigation, and response — and feeds risk into compliance.

5. Wipe vs Retire?

Wipe factory-resets the whole device (corporate); Retire removes only corporate data/management (BYOD).

Scenario-Based Questions

1. Push corporate Wi-Fi and certificates to all laptops.

Create Wi-Fi and certificate (SCEP/PKCS) configuration profiles in the Settings Catalog/templates and assign to the Windows device group.

2. Harden 2,000 laptops fast.

Deploy the Windows security baseline to a pilot ring, validate, then roll out fleet-wide, layering antivirus, firewall, and ASR.

3. Automatically contain a compromised device.

Onboard to Defender for Endpoint, add a machine-risk rule to compliance, and require compliant devices in Conditional Access so high-risk devices are blocked.

4. Patch safely at scale.

Use WUfB rings (Pilot → Broad) with staggered deferrals and deadlines, feature profiles for version control, Expedite for emergencies, or Windows Autopatch.

Troubleshooting Questions

1. A configuration profile shows Conflict. Cause?

Two profiles set the same value differently; the conflicting setting isn't applied. Use per-setting status to find and resolve it.

2. A Windows device is Entra-joined but has no Intune policy. Cause?

Automatic enrollment (MDM scope) not set or missing Intune license; confirm both and sync.

3. A remote wipe is stuck Pending. Why?

The device is offline; the command runs at its next contact with Intune.

Practical Administrator Questions

1. How do you delegate help-desk access to only their devices?

Assign a least-privilege RBAC role (e.g., Help Desk Operator) with scope tags limiting visible objects.

2. How do you auto-fix a recurring device issue?

Deploy a proactive remediation with detection and remediation scripts.

3. How do you stop Office macro malware?

Enable ASR rules (Audit then Block) that stop Office apps from spawning child processes/creating executable content.

Production Environment Questions

1. Design endpoint management for a mixed Windows/iOS/Android fleet.

Autopilot + automatic enrollment for Windows, Apple ADE for corporate Apple, fully managed/work profile for Android, MAM for BYOD, baselines + Defender for security, WUfB rings for patching, all from Intune.

2. How do you prove and improve device security posture?

Use Defender for Endpoint Secure Score and TVM, deploy baselines and ASR, and track Endpoint analytics; export to Power BI for leadership.

3. How do you migrate from Configuration Manager?

Enable co-management and shift workloads (compliance, update, protection) to Intune ring by ring, routing new devices through Autopilot.

Summary

Expect deep questions on Intune fundamentals, MDM/MAM, configuration profiles vs Group Policy, endpoint security and Defender for Endpoint, remote actions, monitoring, and update management. Answer with the business goal, the specific feature, and the exact portal path or tool.

12. Section 3 — Troubleshooting Reference

Introduction

A symptom-first reference for the management, maintenance, and protection issues you'll meet most. Each entry follows Symptoms → Root Cause → Steps → Resolution.

1. Policies/apps not reaching a device

Symptoms: Device enrolled but missing settings/apps.

Root Cause: Not assigned to the device's group, licensing gap, wrong platform, or not synced.

Steps: Verify assignments and platform; confirm Intune license; force a sync.

Resolution: Fix assignment/licensing; content arrives on next check-in.

2. Configuration profile conflict

Symptoms: Profile shows Conflict; setting not applied.

Root Cause: Two profiles set the same value differently.

Steps: Open per-setting status to find the duplicate; remove or align it.

Resolution: One authority per setting; conflict clears.

3. Windows device won't enroll

Symptoms: Entra-joined but not in Intune.

Root Cause: MDM scope unset, no license, or enrollment restriction blocking.

Steps: Confirm automatic enrollment scope and license; check restrictions; `dsregcmd /status`.

Resolution: Enable scope, assign license, adjust restrictions.

4. Apple ADE devices enroll generically

Symptoms: No corporate setup on iPhones/iPads.

Root Cause: Device not assigned to Intune in ABM, expired token, or unassigned enrollment profile.

Steps: Verify ABM assignment; renew tokens; confirm profile assignment; sync.

Resolution: Assign device + profile and renew tokens.

5. Security baseline breaks an app/login

Symptoms: App fails or users can't sign in after a baseline.

Root Cause: A hardened setting is too strict.

Steps: Find the offending setting via per-setting status; adjust it; always pilot baselines first.

Resolution: Tune the baseline; re-pilot before broad rollout.

6. Defender onboarding / risk not working

Symptoms: Devices missing in Defender portal, or risk doesn't block access.

Root Cause: Connector off, EDR policy unassigned, licensing/connectivity gaps, or compliance/CA not wired to risk.

Steps: Enable the Intune–MDE connector; assign EDR onboarding; add machine risk to compliance; require compliant in CA.

Resolution: Complete onboarding and wire risk → compliance → CA.

7. Remote action pending

Symptoms: Wipe/lock stays Pending.

Root Cause: Device offline.

Steps: Confirm the command is queued; it runs at next device contact.

Resolution: Action completes when the device reconnects.

8. Windows Update not installing

Symptoms: Devices unpatched past the deadline.

Root Cause: Conflicting rings, active pause, connectivity, or low disk space.

Steps: Resolve ring conflicts; lift pauses; check connectivity/disk; review WUfB reports.

Resolution: Single ring authority, free space, force sync.

9. Stale monitoring data

Symptoms: Reports outdated or device missing.

Root Cause: No recent sync, or analytics not enabled.

Steps: Confirm sync; enable Endpoint analytics; allow telemetry time.

Resolution: Data populates after sync/enablement.

Best Practices

- Start most investigations by checking **assignment scope** and **sync state**.
- Use **per-setting status** for profile/baseline issues.

- Wire **Defender risk → compliance → Conditional Access** end to end.
- Match **remote actions to ownership** (Wipe vs Retire).
- Keep **Apple tokens** current and **update rings** non-conflicting.

Summary

Most Section 3 issues trace to assignment/scope gaps, missing sync, licensing, profile conflicts, incomplete Defender integration, offline devices, or expired Apple tokens. A symptom-first approach plus per-setting status and the Intune monitoring views resolves the majority quickly.

End of Section 3 — Manage, Maintain, and Protect Devices.
Next: *Section 4 — Manage Applications* (application deployment, Microsoft Store apps, Win32 apps, Microsoft 365 Apps deployment, app protection policies, app configuration policies, application updates, and application troubleshooting), followed by the final reference appendix (exam strategy, cheat sheets, the 50 interview questions, and more). After Section 4, all parts will be merged into a single continuous master PDF.

MICROSOFT CERTIFICATION

MD-102

Microsoft Endpoint Administrator

Section 4 — Manage Applications

A Practical, Interview-Focused Enterprise Study Guide

App Deployment · Microsoft Store Apps · Win32 Apps · Microsoft 365 Apps

App Protection Policies · App Configuration Policies · Updates · Troubleshooting

Real-World Administration • Troubleshooting • Hands-On Labs • Interview Preparation

Section 4 Contents

1. **Application Deployment**
2. **Microsoft Store Apps**
3. **Win32 Apps**
4. **Microsoft 365 Apps Deployment**
5. **App Protection Policies**
6. **App Configuration Policies**
7. **Application Updates**
8. **Application Troubleshooting**
9. **Section 4 — Consolidated Interview Preparation**
10. **Section 4 — Troubleshooting Reference**

How to use this section: Applications are how users actually get work done, so deploying, protecting, and updating them is core endpoint-admin work. Each topic keeps the fixed structure (concepts, architecture, exact portal navigation, step-by-step configuration, scenarios, enterprise examples, troubleshooting, best practices, security, PowerShell, interview Q&A, exam notes, hands-on lab, summary, mini quiz). Build every lab in a trial Intune tenant.

1. Application Deployment

Introduction

Application deployment is how Intune gets the software users need onto their devices — automatically, on demand, or made available in a self-service catalog. Instead of technicians installing apps by hand, you add an app once, assign it to a group, and Intune delivers it to every targeted device. This is one of the most frequent daily tasks for an endpoint administrator.

Why This Topic Is Important

Users are only productive if they have their apps. Efficient, reliable app deployment reduces help-desk load, ensures consistency, and is a core exam and interview area. Understanding **app types** and **assignment intents** is fundamental to everything else in this section.

Key Concepts

- **App types:** Microsoft Store apps, Win32 (line-of-business) apps, Microsoft 365 Apps, LOB packages (MSI/APPX), web links, and built-in/store apps for iOS/Android.
- **Assignment intents:** *Required* (auto-installs), *Available for enrolled devices* (shows in Company Portal for users to install), and *Uninstall* (removes the app).
- **Company Portal:** The self-service app where users browse and install Available apps.
- **Detection rules:** How Intune decides whether an app is already installed (file, registry, MSI code, or script).
- **Dependencies & supersedence:** Apps that must install first, and replacing an old app version with a new one.
- **Targeting:** Assign to user or device groups (include/exclude).

Assignment intents

Intent	Behavior	Typical use
Required	Installs automatically on targeted devices/users	Mandatory corporate apps (security agent, Office)
Available for enrolled devices	Appears in Company Portal for optional install	Self-service / optional productivity apps
Uninstall	Removes the app from targeted devices	Retiring or replacing software

Architecture / Workflow

1. Admin **adds the app** to Intune (chooses the app type and uploads/links the package).
2. Admin configures **detection rules**, dependencies, and any requirements.
3. The app is **assigned** to groups with an intent (Required/Available/Uninstall).

4. At sync, the Intune agent installs/removes the app and evaluates detection.
5. The device reports **install status**; admins monitor results.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → All apps → Add
```

```
Intune Admin Center → Apps → All apps → (app) → Properties → Assignments
```

```
Intune Admin Center → Apps → Monitor → App install status
```

```
Intune Admin Center → Apps → (platform) → (filtered app lists)
```

Step-by-Step Configuration

A. Add and assign an app

1. Go to *Apps* → *All apps* → *Add* and select the app type.
2. Provide package/details, logo, and description.
3. Configure **detection rules** and requirements (for Win32/LOB).
4. On the **Assignments** tab, add the target group and choose **Required** or **Available**.
5. Create; monitor under *Apps* → *Monitor* → *App install status*.

B. Make an app self-service

1. Assign the app as **Available for enrolled devices** to a user group.
2. Users open **Company Portal** and install on demand.

Real-World Scenario

Scenario: Department rollout

The Design team needs Adobe tools and the Sales team needs a CRM client. The admin assigns each app as **Required** to the respective department dynamic group, so members get the right software automatically — no tickets, no manual installs.

Scenario: Self-service catalog

Optional tools (a screen recorder, a PDF editor) are assigned as **Available**. Employees install them from Company Portal when needed, reducing requests to IT.

Enterprise Deployment Example

An enterprise standardizes app delivery: a baseline set of Required apps (security agent, Microsoft 365 Apps, VPN) goes to all devices; role-specific apps target department dynamic groups; and a curated Company Portal offers optional Available apps. Supersedence keeps versions current, and app install reports drive remediation of failures.

Troubleshooting

App not installing

Symptoms: Required app shows Failed or Pending; not on device.

Root Cause: Wrong/over-strict detection rule, unmet requirement, dependency missing, or device not synced.

Steps: Review App install status for the error code; verify detection rules and requirements; confirm dependencies; force sync.

Resolution: Correct detection/requirements/dependency; app installs on next check-in.

App shows "Installed" but isn't (or vice versa)

Symptoms: Status doesn't match reality.

Root Cause: Detection rule doesn't accurately match the installed state.

Steps: Adjust the detection rule (correct file path/version, registry key, or MSI code).

Resolution: Accurate detection rule; status reflects reality.

Best Practices

- Use **Required** for mandatory apps, **Available** for optional self-service.
- Get **detection rules** right — most app problems are detection problems.
- Target with **dynamic groups** for accurate, automatic delivery.
- Use **supersedence** to replace old versions cleanly.
- Monitor **App install status** and remediate failures promptly.

Security Considerations

App deployment is a powerful channel — only deploy trusted, vetted software, and restrict who can add/assign apps via RBAC. Available apps in Company Portal should be curated to prevent unwanted software. For BYOD, prefer app-level protection (covered later) so corporate apps and data stay contained.

PowerShell Commands

```
# List apps and assignments via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementMobileApp | Select-Object DisplayName, Id

# App install status is available through the app's device statuses
Get-MgDeviceAppManagementMobileApp -MobileAppId <id>
```

Interview Questions and Answers

BEGINNER **What is application deployment in Intune?**

Adding an app once and assigning it to groups so Intune delivers it automatically or via self-service to targeted devices.

BEGINNER **Name the assignment intents.**

Required (auto-install), Available for enrolled devices (Company Portal self-service), and Uninstall.

INTERMEDIATE **What are detection rules and why do they matter?**

They tell Intune whether an app is already installed (file, registry, MSI code, or script); wrong rules cause incorrect install status and repeated install attempts.

INTERMEDIATE **How do users install optional apps themselves?**

From the Company Portal, for apps assigned as Available for enrolled devices.

ADVANCED **A Required app shows Failed. How do you troubleshoot?**

Check App install status for the error code, verify detection rules and requirements, confirm dependencies, and force a sync; most failures are detection or requirement issues.

MD-102 Exam Notes

> **Exam Focus**
>
> Know the **intents**: Required (auto), Available (Company Portal), Uninstall. **Detection rules** determine install status. Apps target **user or device groups**. **Supersedence** replaces versions; **dependencies** install prerequisites first. Monitor via **App install status**.

Hands-On Lab

> **Lab 1.1 — Deploy apps**
>
> 1. Add a simple app and assign it as Required to a test device group. 2. Add another app as Available to a user group. 3. Configure detection rules for a Win32/LOB app. 4. Open Apps → Monitor → App install status and review results. 5. Install an Available app from Company Portal as a test user.

Summary

Application deployment lets Intune deliver software automatically (Required), via self-service (Available), or remove it (Uninstall). Accurate detection rules, dynamic-group targeting, supersedence, and install monitoring are the keys to reliable delivery, with Company Portal providing the user-facing catalog.

Mini Quiz

1. Which intent installs an app automatically?
2. Where do users install optional apps?
3. What determines whether Intune thinks an app is installed?
4. What replaces an old app version cleanly?
5. Where do you monitor install results?

Answers: 1) Required. 2) Company Portal (Available apps). 3) Detection rules. 4) Supersedence. 5) Apps → Monitor → App install status.

2. Microsoft Store Apps

Introduction

Microsoft Store apps are applications delivered through the Microsoft Store and added to Intune via the **new Microsoft Store app (new)** experience, which integrates the Store catalog directly into Intune. Store apps update themselves and are simple to deploy — you search the catalog, pick the app, and assign it. They're ideal for common, well-known applications that don't need complex packaging.

Why This Topic Is Important

Store apps are the quickest way to deploy mainstream software with minimal effort and built-in updates. Knowing when to use a Store app versus a Win32 package is a recurring exam and real-world decision, so the comparison matters.

Key Concepts

- **Microsoft Store app (new):** The current method that searches the Store catalog from within Intune and supports both UWP and packaged Win32 apps available in the Store.
- **Automatic updates:** Store apps generally update themselves, reducing maintenance.
- **Winget backbone:** The new Store experience leverages the Windows Package Manager (winget) catalog.
- **Assignment intents:** Same Required/Available/Uninstall model as other apps.
- **System vs user context:** Store apps can install in device (system) or user context depending on the app.

Microsoft Store Apps vs Win32 Apps — Comparison

Aspect	Microsoft Store Apps	Win32 Apps
Packaging	None — pick from Store catalog	Wrap into .intunewin with Content Prep Tool
Updates	Automatic via the Store	You manage updates (supersedence)
Customization	Limited	Full control (install commands, scripts, detection)
Best for	Common, mainstream apps	Complex/legacy/custom line-of-business apps
Effort	Very low	Higher (packaging + rules)
Detection rules	Handled automatically	You define them

Architecture / Workflow

1. Admin chooses **Microsoft Store app (new)** and searches the catalog.
2. Intune pulls the app metadata and package reference from the Store.
3. The app is **assigned** to groups with an intent.

4. Devices install from the Store; the Store handles updates automatically.
5. Install status is reported back to Intune.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → All apps → Add → App type: Microsoft Store app (new)
```

```
Intune Admin Center → Apps → All apps → (Store app) → Assignments
```

Step-by-Step Configuration

1. Go to *Apps* → *All apps* → *Add*; App type = **Microsoft Store app (new)**.
2. Click *Search the Microsoft Store app (new)* and find the app.
3. Select it; review the auto-populated details and install behavior (system/user).
4. Assign to a group as Required or Available.
5. Create and monitor install status.

Real-World Scenario

Scenario: Quick rollout of a common tool

The company standardizes on a popular note-taking app. Rather than packaging it, the admin adds it as a **Microsoft Store app (new)**, assigns it as Required to all users, and lets the Store keep it updated automatically — minutes of work, zero ongoing maintenance.

Enterprise Deployment Example

An enterprise uses Store apps for mainstream software (browsers, utilities, common productivity tools) to minimize packaging effort and offload updates to the Store, reserving Win32 packaging for complex or legacy line-of-business apps. A curated set of Store apps is offered as Available in Company Portal.

Troubleshooting

Store app won't install

Symptoms: App fails to install from the Store.

Root Cause: Store/winget access blocked, device offline, or app not available for the platform/region.

Steps: Confirm the device can reach the Store/winget; verify assignment and platform support; force sync.

Resolution: Restore Store access/assignment; app installs on next check-in.

Best Practices

- Use **Store apps** for common software to minimize effort and gain automatic updates.
- Reserve **Win32** for complex, legacy, or highly customized apps.

- Curate the **Available** Store catalog in Company Portal.
- Verify the app supports your platform/context before broad assignment.

Security Considerations

Store apps come from a trusted, signed source and update automatically, which improves security hygiene. Still, curate which apps are available to avoid unwanted software, and ensure devices can reach the Store securely. Blocking the consumer Store while allowing managed Store deployment is a common balance.

PowerShell Commands

```
# Winget underpins the new Store experience; inspect locally
winget search "App Name"
winget list

# Store apps appear among Intune apps via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementMobileApp | Select-Object DisplayName
```

Interview Questions and Answers

BEGINNER **What is a Microsoft Store app in Intune?**

An app added from the Microsoft Store catalog via the "Microsoft Store app (new)" type — no packaging needed, with automatic updates.

INTERMEDIATE **When do you choose a Store app over a Win32 app?**

For common, mainstream apps where you want minimal effort and automatic updates; Win32 is for complex, legacy, or highly customized apps.

ADVANCED **What technology underpins the new Store app experience?**

The Windows Package Manager (winget) catalog, which the new Microsoft Store app type leverages.

MD-102 Exam Notes

Exam Focus

Microsoft Store app (new) = search the catalog in Intune, no packaging, **automatic updates**, backed by **winget**. Detection is handled automatically. Choose Store apps for mainstream software; Win32 for complex/custom apps.

Hands-On Lab

Lab 2.1 — Deploy a Store app

1. Apps → Add → Microsoft Store app (new). 2. Search for a common app and select it. 3. Note the install context (system/user). 4. Assign as Required to a test group. 5. Confirm install on a device and verify it updates from the Store.

Summary

Microsoft Store apps deploy mainstream software with minimal effort and automatic updates via the new winget-backed Store experience in Intune. They're ideal for common apps, while Win32 packaging is reserved for complex or legacy software. Curate the catalog and ensure Store connectivity for reliable delivery.

Mini Quiz

1. Which app type needs no packaging?
2. How do Store apps get updated?
3. What package manager backs the new Store experience?
4. When would you use Win32 instead?
5. Who handles detection rules for Store apps?

Answers: 1) Microsoft Store app (new). 2) Automatically via the Store. 3) winget (Windows Package Manager). 4) For complex, legacy, or customized apps. 5) Intune, automatically.

3. Win32 Apps

Introduction

Win32 apps are traditional Windows desktop applications (EXE/MSI installers and their scripts) packaged for Intune. You wrap the installer into a **.intunewin** file using the **Microsoft Win32 Content Prep Tool**, then define install/uninstall commands, detection rules, requirements, and dependencies. Win32 deployment gives you full control and is the go-to method for complex, legacy, or highly customized line-of-business software.

Why This Topic Is Important

Most real enterprises have custom and legacy apps that aren't in the Store. Win32 packaging is the most powerful and most tested app-deployment method in MD-102, and packaging skills (the Content Prep Tool, detection rules, install commands) are frequently probed in interviews.

Key Concepts

- **.intunewin file:** The packaged app produced by the Content Prep Tool, uploaded to Intune.
- **Microsoft Win32 Content Prep Tool:** Command-line tool (`IntuneWinAppUtil.exe`) that wraps a source folder + setup file into `.intunewin`.
- **Install/uninstall commands:** The exact command lines Intune runs (e.g., `msiexec /i app.msi /qn`).
- **Detection rules:** File, registry, MSI product code, or custom script that confirms successful install.
- **Requirements:** Conditions a device must meet (OS, architecture, disk space) before install.
- **Dependencies:** Apps that must be installed first.
- **Supersedence:** Replace/upgrade an existing app, optionally uninstalling the old version.
- **Install context:** System (all users) or User.
- **Return codes:** Map exit codes to success/soft reboot/hard reboot/retry.

Architecture / Workflow

1. Gather the installer + any scripts in a **source folder**.
2. Run the **Content Prep Tool** to produce a `.intunewin` file.
3. In Intune, add a **Windows app (Win32)** and upload the `.intunewin`.
4. Define **install/uninstall commands**, **detection rules**, requirements, dependencies, and return codes.
5. Assign to groups (Required/Available/Uninstall).
6. The Intune Management Extension installs the app and evaluates detection; status is reported.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → All apps → Add → App type: Windows app (Win32)
```

```
Intune Admin Center → Apps → (Win32 app) → Properties (Program, Detection,
Requirements, Dependencies, Supersedence)
```

```
Intune Admin Center → Apps → Monitor → App install status
```

Step-by-Step Configuration

A. Package the app

```
# Run the Content Prep Tool (download from Microsoft's GitHub)
IntuneWinAppUtil.exe -c C:\Source -s setup.exe -o C:\Output
# Produces setup.intunewin in C:\Output
```

B. Add to Intune

1. Go to *Apps* → *Add* → *Windows app (Win32)*; upload the `.intunewin`.
2. Set **install command** (e.g., `setup.exe /silent`) and **uninstall command**.
3. Choose install context (System/User) and map **return codes**.
4. Configure **requirements** (OS, architecture, min disk/RAM).
5. Configure a **detection rule** (e.g., MSI product code, file version, or registry key).
6. Add **dependencies** / **supersedence** if needed.
7. Assign to groups and create.

Real-World Scenario

Scenario: Legacy LOB app

A company has a custom accounting app with a silent EXE installer and a prerequisite runtime. The admin packages it into a `.intunewin`, sets the runtime as a **dependency**, defines a registry-based detection rule, and deploys it as Required to the Finance group — fully automated, with the prerequisite handled first.

Scenario: Version upgrade

A new version of an internal tool is released. The admin packages it and configures **supersedence** to uninstall the old version and install the new one, cleanly upgrading the fleet.

Enterprise Deployment Example

An enterprise maintains a packaging pipeline: source installers are wrapped with the Content Prep Tool, versioned in a repository, and deployed via Win32 with strict detection rules and requirement filters. Supersedence chains manage upgrades, dependencies enforce prerequisites, and app install reports feed a remediation queue. Complex apps that the Store can't handle all flow through this Win32 process.

Troubleshooting

Win32 app installation failure

Symptoms: App shows Failed with an error code.

Root Cause: Wrong install command/switches, unmet requirement, missing dependency, or incorrect return-code mapping.

Steps: Review the error code and the IME log (`IntuneManagementExtension.log`); verify the silent install switches by testing the command locally; confirm requirements/dependencies and return codes.

Resolution: Fix the command/requirements/return codes; redeploy.

App installs but reports Failed (or reinstalls repeatedly)

Symptoms: Status wrong; app keeps trying to install.

Root Cause: Detection rule doesn't match the installed state.

Steps: Verify the detection rule against what the installer actually creates (correct path/version/registry/MSI code).

Resolution: Correct the detection rule so status reflects reality.

Best Practices

- Always test the **silent install/uninstall commands** locally before packaging.
- Use precise **detection rules** — the most common cause of Win32 issues.
- Set realistic **requirements** and correct **return-code** mappings.
- Use **dependencies** for prerequisites and **supersedence** for upgrades.
- Check the **IME log** (`IntuneManagementExtension.log`) for failures.

Security Considerations

Win32 apps run installers with system privileges, so only package trusted, vetted software and restrict who can add apps. Validate sources and signatures, avoid bundling risky scripts, and keep packages in a controlled repository. A malicious or misconfigured Win32 package can affect every targeted device.

PowerShell Commands

```
# Example MSI silent install/uninstall commands used in Win32 apps
msiexec /i "App.msi" /qn        # install
msiexec /x "{PRODUCT-GUID}" /qn # uninstall

# Find an installed MSI product code for detection rules
Get-WmiObject Win32_Product | Select-Object Name, IdentifyingNumber

# Win32 app install logs (on the device)
# C:\ProgramData\Microsoft\IntuneManagementExtension\Logs\IntuneManagementExtension.log
```

Interview Questions and Answers

BEGINNER **What is a Win32 app in Intune?**

A traditional Windows desktop app packaged into a .intunewin file and deployed with install commands, detection rules, and requirements.

BEGINNER **What tool creates the .intunewin file?**

The Microsoft Win32 Content Prep Tool (IntuneWinAppUtil.exe).

INTERMEDIATE **What are detection rules and why do they matter?**

They confirm whether the app is installed (file, registry, MSI code, or script); incorrect rules cause wrong status and repeated install attempts — the most common Win32 problem.

INTERMEDIATE **How do you handle a prerequisite runtime?**

Configure it as a dependency so it installs before the main app.

ADVANCED **A Win32 app fails to install — what's your process?**

Check the error code and the IME log, test the silent install command locally, verify requirements/dependencies and return-code mappings, and confirm the detection rule; fix and redeploy.

MD-102 Exam Notes

> **Exam Focus**
>
> Win32 = **.intunewin** via the **Content Prep Tool**. Configure **install/uninstall commands**, **detection rules**, **requirements**, **dependencies**, **supersedence**, **return codes**, and **install context**. Troubleshoot with the **IntuneManagementExtension.log**. Detection rules are the usual culprit.

Hands-On Lab

> **Lab 3.1 — Package and deploy a Win32 app**
>
> 1. Download the Content Prep Tool and wrap a sample installer into a `.intunewin`. 2. Add it as a Windows app (Win32). 3. Set silent install/uninstall commands and a detection rule (MSI code or file version). 4. Add a requirement (OS/architecture). 5. Assign as Required, then review App install status and the IME log on a device.

Summary

Win32 apps give full control over deploying complex, legacy, and custom Windows software. You package with the Content Prep Tool into a .intunewin, then configure install commands, detection rules, requirements, dependencies, supersedence, and return codes. Detection rules are the most common failure point, and the IME log is the key troubleshooting tool.

Mini Quiz

1. What file format is a packaged Win32 app?
2. What tool creates it?
3. What confirms an app is installed?

4. What handles prerequisites?
5. Which log helps troubleshoot Win32 installs?

Answers: 1) .intunewin. 2) Microsoft Win32 Content Prep Tool (IntuneWinAppUtil.exe). 3) Detection rules. 4) Dependencies. 5) IntuneManagementExtension.log.

4. Microsoft 365 Apps Deployment

Introduction

Microsoft 365 Apps (Word, Excel, PowerPoint, Outlook, Teams, etc.) have a dedicated, streamlined deployment experience in Intune. Instead of packaging Office yourself, you use the built-in **Microsoft 365 Apps** app type, configure which apps and settings you want via a simple wizard, and assign it — Intune handles the Click-to-Run installation and updates.

Why This Topic Is Important

Office is on virtually every corporate device, so deploying and configuring it correctly is essential daily work. The dedicated app type, update channels, and configuration options are tested on MD-102 and commonly discussed in interviews.

Key Concepts

- **Microsoft 365 Apps (app type):** The built-in Intune experience for deploying Office Click-to-Run.
- **Click-to-Run:** Microsoft's streaming install/update technology for Office.
- **Update channels:** Current Channel, Monthly Enterprise Channel, Semi-Annual Enterprise Channel — controlling how often feature updates arrive.
- **Architecture:** 32-bit vs 64-bit selection.
- **App selection:** Choose which Office apps to include/exclude.
- **Languages & settings:** Configure languages, EULA acceptance, shared computer activation.
- **Shared computer activation:** For multi-user/VDI devices.

Architecture / Workflow

1. Admin adds a **Microsoft 365 Apps** app and configures apps, architecture, update channel, and settings via the wizard.
2. The app is **assigned** to groups (usually Required).
3. Devices install Office via **Click-to-Run**.
4. Office updates automatically per the chosen **update channel**.
5. Install status is reported to Intune.

Update channels

Channel	Feature update cadence	Best for
Current Channel	As soon as ready (frequent)	Users wanting the latest features
Monthly Enterprise Channel	Once a month (predictable)	Most enterprises — balance of features + stability
Semi-Annual Enterprise Channel	Twice a year	Highly change-averse / regulated environments

Microsoft Portal Navigation

```
Intune Admin Center → Apps → All apps → Add → App type: Microsoft 365 Apps (Windows 10 and later)
```

```
Intune Admin Center → Apps → (Microsoft 365 Apps) → Assignments
```

Step-by-Step Configuration

1. Go to *Apps → Add → Microsoft 365 Apps (Windows 10 and later)*.
2. Use **Configuration designer** (or import XML) to choose apps, **architecture** (64-bit recommended), and **update channel** (Monthly Enterprise recommended).
3. Set languages, EULA acceptance, and shared computer activation if needed.
4. Assign as Required to a device group.
5. Create and monitor install status.

Real-World Scenario

Scenario: Standardized Office

A company wants 64-bit Office without Publisher/Access on the Monthly Enterprise Channel. The admin configures the Microsoft 365 Apps deployment accordingly and assigns it as Required to all devices — every machine gets a consistent, auto-updating Office build with no manual setup.

Enterprise Deployment Example

An enterprise deploys 64-bit Microsoft 365 Apps on the Monthly Enterprise Channel to most users for predictable monthly updates, uses Semi-Annual Enterprise Channel for a regulated business unit, and enables shared computer activation on its VDI hosts. Update behavior is governed centrally, keeping Office consistent and current across the fleet.

Troubleshooting

Microsoft 365 Apps won't install

Symptoms: Office fails to install or conflicts.

Root Cause: An existing Office (different bitness/MSI) conflicts, connectivity issue, or assignment not applied.

Steps: Check for a conflicting Office install/architecture; ensure removal of older MSI Office; verify connectivity to Office CDN and assignment; review install status.

Resolution: Remove conflicts/match architecture; Office installs cleanly.

Office not on the expected channel

Symptoms: Wrong update cadence/version.

Root Cause: Channel set incorrectly or overridden by another policy.

Steps: Confirm the channel in the app config; check for a conflicting update policy.

Resolution: Set the intended channel; remove conflicts.

Best Practices

- Standardize on **64-bit** Office unless a legacy add-in requires 32-bit.
- Use **Monthly Enterprise Channel** for most users (predictable monthly updates).
- Remove conflicting/older Office (MSI) before deploying Click-to-Run.
- Enable **shared computer activation** for VDI/multi-user devices.
- Keep the app selection lean (exclude unused apps).

Security Considerations

Keeping Office current via an enterprise update channel closes security vulnerabilities promptly. Pair Microsoft 365 Apps deployment with Administrative Templates/Settings Catalog to enforce Office security policies (macro restrictions, protected view) and with app protection for mobile Office apps.

PowerShell Commands

```
# Check the installed Office update channel / version (Click-to-Run)
Get-ItemProperty "HKLM:\SOFTWARE\Microsoft\Office\ClickToRun\Configuration" |
  Select-Object VersionToReport, CDNBaseUrl, UpdateChannel

# Microsoft 365 Apps appears among Intune apps via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementMobileApp | Select-Object DisplayName
```

Interview Questions and Answers

BEGINNER **How do you deploy Office with Intune?**

Use the built-in Microsoft 365 Apps app type, configure apps/architecture/channel via the wizard, and assign it — Intune handles the Click-to-Run install.

BEGINNER **What is Click-to-Run?**

Microsoft's streaming install and update technology for Office.

INTERMEDIATE **Which update channel suits most enterprises and why?**

Monthly Enterprise Channel — predictable monthly feature updates balancing new features with stability.

INTERMEDIATE **What is shared computer activation for?**

Allowing Office activation on multi-user or VDI devices where many users sign in.

ADVANCED **Office won't install on some devices — common cause?**

A conflicting existing Office install or architecture mismatch (e.g., old 32-bit MSI Office); remove it before deploying Click-to-Run.

MD-102 Exam Notes

Exam Focus

Use the **Microsoft 365 Apps** app type (Click-to-Run). Know **update channels** (Current, Monthly Enterprise, Semi-Annual Enterprise), **architecture** (64-bit default), **shared computer activation** for VDI, and that conflicting/old MSI Office must be removed first.

Hands-On Lab

Lab 4.1 — Deploy Microsoft 365 Apps

1. Apps → Add → Microsoft 365 Apps. 2. In the configuration designer, choose 64-bit, exclude an app, and set Monthly Enterprise Channel. 3. Assign as Required to a test group. 4. Confirm Office installs on a device. 5. Verify the update channel via the registry command.

Summary

Microsoft 365 Apps deployment uses a dedicated Intune app type and Click-to-Run to install and auto-update Office. Administrators choose apps, architecture (64-bit default), and an update channel (Monthly Enterprise for most), enable shared computer activation for VDI, and remove conflicting older Office before deploying.

Mini Quiz

1. What install technology does Office use?
2. Which channel suits most enterprises?
3. What architecture is the default recommendation?
4. What feature supports multi-user/VDI activation?
5. What must be removed before deploying Click-to-Run Office?

Answers: 1) Click-to-Run. 2) Monthly Enterprise Channel. 3) 64-bit. 4) Shared computer activation. 5) Conflicting/older (MSI) Office installs.

5. App Protection Policies

Introduction

App protection policies (APP) — the core of **Mobile Application Management (MAM)** — protect corporate *data inside apps* rather than managing the whole device. They control what happens to company data within apps like Outlook and Teams: blocking copy-paste to personal apps, requiring a PIN to open the app, encrypting app data, and selectively wiping only corporate data. This is the key technology for securing **BYOD** without taking over personal devices.

Why This Topic Is Important

BYOD is everywhere, and app protection is how you secure corporate data on devices you don't fully manage — a major exam and real-world topic. The MAM-vs-MDM distinction and the "protect data without managing the device" pattern come up constantly.

Key Concepts

- **MAM (Mobile Application Management):** Managing corporate apps/data, not the device.
- **App protection policy:** Rules applied to managed apps (data relocation, access, conditional launch).
- **Data relocation controls:** Restrict cut/copy/paste, save-as, and "open in" to keep data in managed apps.
- **Access requirements:** App PIN, biometric, or work-account credentials to open the app.
- **Conditional launch:** Conditions (jailbreak/root detection, min OS, max PIN attempts) that block or wipe.
- **Selective wipe:** Removing only corporate app data, leaving personal data intact.
- **MAM-WE (without enrollment):** App protection applied to apps on unenrolled (BYOD) devices.
- **App protection + Conditional Access:** CA can require an app protection policy for access.

Architecture / Workflow

1. Admin creates an **app protection policy** per platform (iOS/Android) and selects managed apps.
2. Data relocation, access, and conditional launch settings are configured.
3. The policy is **assigned** to user groups.
4. When a user opens a managed app with their work account, the policy enforces PIN, encryption, and data controls.
5. **Conditional Access** can require an app protection policy; admins can **selectively wipe** corporate data when needed.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → App protection policies → Create policy
```

```
Intune Admin Center → Apps → App protection policies → (policy) → Assignments
```

```
Intune Admin Center → Apps → Monitor → App protection status
```

```
Intune Admin Center → Apps → All apps → (app) → … (Wipe app data / selective wipe via app protection)
```

Step-by-Step Configuration

1. Go to *Apps* → *App protection policies* → *Create policy*; choose the platform.
2. Select **managed apps** (e.g., Outlook, Teams, Edge).
3. **Data protection:** restrict cut/copy/paste to policy-managed apps; block backup; restrict save-as.
4. **Access requirements:** require an app PIN/biometric; require corporate credentials.
5. **Conditional launch:** block jailbroken/rooted devices; set max PIN attempts → wipe; set min OS.
6. Assign to a user group; monitor under *App protection status*.

Real-World Scenario

Scenario: Secure BYOD email

Employees use personal phones for Outlook. An app protection policy requires a PIN to open Outlook, blocks copy-paste of corporate content into personal apps, and encrypts the app's data. When someone leaves, IT issues a **selective wipe** that removes only the corporate Outlook data — personal photos and apps are untouched.

Scenario: Block data exfiltration

A policy prevents "save as" to personal cloud storage and restricts "open in" to managed apps only, stopping corporate documents from leaking into unmanaged apps.

Enterprise Deployment Example

An enterprise secures BYOD entirely with MAM: app protection policies on Outlook, Teams, Edge, and Office mobile enforce PIN, encryption, and strict data relocation; conditional launch blocks rooted devices and wipes after failed PIN attempts; and Conditional Access requires an app protection policy for mobile access. No personal device is enrolled, yet corporate data is fully contained — and removable via selective wipe.

Troubleshooting

App protection policy not applying

Symptoms: No PIN prompt; data controls absent.

Root Cause: Policy not assigned to the user, app not a policy-managed app, user not signed in with the work account, or no Intune license.

Steps: Confirm assignment to the user; ensure the app is in the managed list and the user signs in with the corporate account; verify licensing; check App protection status.

Resolution: Assign correctly and use the work account; policy enforces on next app launch.

Users blocked unexpectedly by conditional launch

Symptoms: App blocks or wipes when it shouldn't.

Root Cause: Conditional launch thresholds too strict (min OS, PIN attempts).

Steps: Review conditional launch settings against the real device fleet; relax thresholds appropriately.

Resolution: Tune conditional launch to balance security and usability.

Best Practices

- Use **app protection (MAM)** for BYOD to secure data without managing the device.
- Require an **app PIN** and **encryption**; restrict data relocation to managed apps.
- Configure **conditional launch** (block rooted, max PIN attempts) sensibly.
- Pair with **Conditional Access** requiring an app protection policy.
- Use **selective wipe** for clean offboarding.

Security Considerations

App protection is the privacy-respecting way to secure corporate data on personal devices — it contains and can wipe company data without touching personal content. The strongest pattern is APP + Conditional Access requiring an app protection policy, ensuring only protected app sessions reach data. Tune conditional launch to avoid both data leakage and excessive lockouts.

PowerShell Commands

```
# List app protection policies via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementiOSManagedAppProtection | Select-Object DisplayName
Get-MgDeviceAppManagementAndroidManagedAppProtection | Select-Object DisplayName

# App protection status / wipe requests are managed via the managed app endpoints in
Graph
```

Interview Questions and Answers

BEGINNER **What is an app protection policy?**

A MAM policy that protects corporate data inside managed apps — PIN, encryption, data relocation controls, and selective wipe — without managing the whole device.

BEGINNER **What is selective wipe?**

Removing only corporate data from managed apps, leaving personal data intact.

INTERMEDIATE **Why use app protection for BYOD?**

It secures corporate data and is removable, without enrolling or controlling the user's personal device — respecting privacy.

INTERMEDIATE **What is conditional launch?**

Conditions (root/jailbreak detection, min OS, max PIN attempts) that block app access or trigger a wipe when not met.

ADVANCED **How do you ensure only protected app sessions reach corporate data?**

Combine app protection policies with a Conditional Access policy that requires an app protection policy for access.

MD-102 Exam Notes

Exam Focus

App protection = **MAM**: protect data *in apps*, not the device. Key controls: **data relocation** (copy/paste, save-as), **access** (app PIN), **conditional launch** (root detection, PIN attempts → wipe), and **selective wipe**. Works **without enrollment (MAM-WE)**. Pair with **Conditional Access** (require app protection policy). Ideal for **BYOD**.

Hands-On Lab

Lab 5.1 — Protect app data

1. Create an iOS or Android app protection policy. 2. Select Outlook/Teams as managed apps. 3. Require an app PIN and restrict copy/paste to managed apps. 4. Set conditional launch (block rooted, max PIN attempts → wipe). 5. Assign to a user group and review App protection status; practice a selective wipe on a test account.

Summary

App protection policies (MAM) secure corporate data inside managed apps with PIN, encryption, data-relocation limits, and conditional launch, and allow selective wipe of only corporate data. They protect BYOD without managing the device, work without enrollment (MAM-WE), and are strongest combined with Conditional Access requiring an app protection policy.

Mini Quiz

1. What does an app protection policy protect?

2. What removes only corporate data?
3. What feature blocks rooted devices or wipes after failed PINs?
4. What ownership model is app protection ideal for?
5. What CA control pairs with it?

Answers: 1) Corporate data inside managed apps. 2) Selective wipe. 3) Conditional launch. 4) BYOD / personal devices. 5) Require an app protection policy.

6. App Configuration Policies

Introduction

App configuration policies pre-configure app settings so users don't have to. Instead of each user manually entering a server URL, email setting, or feature toggle, the policy pushes those values into the app automatically. This makes apps work correctly on first launch and enforces consistent, supported configurations.

Why This Topic Is Important

Pre-configuring apps reduces setup friction and support tickets and ensures apps are configured the way IT intends. App configuration is distinct from app *protection* (settings vs data security), a distinction the exam tests.

Key Concepts

- **App configuration policy:** Delivers configuration key/value settings into supported apps.
- **Two delivery types:** *Managed devices* (enrolled/MDM) and *Managed apps* (MAM, via the app protection framework).
- **Configuration keys:** App-specific settings (e.g., a server URL, default account, feature flags) provided by the app vendor.
- **Configuration vs protection:** Configuration sets up the app; protection secures its data.
- **Supported apps:** The app must support app configuration (many Microsoft and managed apps do).

App Protection vs App Configuration (quick contrast)

Aspect	App Protection Policy	App Configuration Policy
Purpose	Secure corporate data in the app	Pre-configure app settings
Examples	PIN, encryption, copy/paste limits, wipe	Server URL, default account, feature toggles
Question it answers	"Is the data safe?"	"Is the app set up correctly?"

Architecture / Workflow

1. Admin creates an app configuration policy (Managed devices or Managed apps).
2. Configuration **keys/values** are entered (from the app's documentation).
3. The policy is **assigned** to groups.
4. The supported app reads the configuration on launch and applies it automatically.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → App configuration policies → Add (Managed devices / Managed apps)
```

```
Intune Admin Center → Apps → App configuration policies → (policy) → Assignments
```

Step-by-Step Configuration

1. Go to *Apps* → *App configuration policies* → *Add*; choose **Managed devices** or **Managed apps**.
2. Select the target app/platform.
3. Enter **configuration keys and values** (per the app vendor's documentation), or use the configuration designer.
4. Assign to a group.
5. Create; verify the app launches pre-configured.

Real-World Scenario

Scenario: Pre-configured email

Instead of asking users to type server details into a mail app, the admin creates an app configuration policy that pushes the account and server settings automatically. Users open the app and it's already set up and ready — no manual steps, no tickets.

Enterprise Deployment Example

An enterprise pre-configures its managed apps fleet-wide: a line-of-business app receives its API endpoint and tenant ID, the mail app gets account defaults, and Edge mobile gets bookmarks and a homepage — all via app configuration policies. Users get correctly configured apps on first launch, reducing onboarding friction across thousands of devices.

Troubleshooting

App configuration not applied

Symptoms: App opens unconfigured despite the policy.

Root Cause: Wrong delivery type (Managed devices vs Managed apps), unsupported app, incorrect keys, or unassigned/unsynced.

Steps: Confirm the app supports app configuration and the correct delivery type; verify keys/values against the vendor docs; check assignment; force sync.

Resolution: Correct delivery type/keys/assignment; app reads config on next launch.

Best Practices

• Match the **delivery type** to the management model (Managed devices = MDM, Managed apps = MAM).

- Use the **vendor's documented keys/values** exactly.
- Combine with **app protection** (configure + secure the same app).
- Test on one device before broad assignment.

Security Considerations

App configuration improves consistency and can enforce supported, safer settings, but it isn't a data-security control — pair it with app protection policies for that. Avoid placing secrets in configuration values where avoidable, and restrict who can edit configuration policies via RBAC.

PowerShell Commands

```
# App configuration policies via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementMobileAppConfiguration | Select-Object DisplayName
Get-MgDeviceAppManagementTargetedManagedAppConfiguration | Select-Object DisplayName
```

Interview Questions and Answers

BEGINNER **What is an app configuration policy?**

A policy that pre-configures app settings (like a server URL or default account) so the app works correctly on first launch.

BEGINNER **How does it differ from an app protection policy?**

Configuration sets up the app's settings; protection secures the app's corporate data.

INTERMEDIATE **What are the two delivery types?**

Managed devices (enrolled/MDM) and Managed apps (MAM via the app protection framework).

ADVANCED **An app ignores its configuration — what do you check?**

Whether the app supports app configuration, the correct delivery type is used, the keys/values match the vendor docs, and the policy is assigned and synced.

MD-102 Exam Notes

Exam Focus

App **configuration** ≠ app **protection**. Two delivery types: **Managed devices (MDM)** and **Managed apps (MAM)**. Uses vendor-defined **keys/values**. The app must **support** app configuration. Often combined with app protection on the same app.

Hands-On Lab

Lab 6.1 — Configure an app

1. Apps → App configuration policies → Add → choose Managed apps or Managed devices. 2. Select a supported app. 3. Enter configuration keys/values (or use the designer). 4. Assign to a user group. 5. Launch the app on a test device and confirm it's pre-configured.

Summary

App configuration policies pre-populate app settings so apps work correctly on first launch, delivered via Managed devices (MDM) or Managed apps (MAM) using vendor-defined keys. They're distinct from app protection (settings vs data security) and are commonly combined with it to both configure and secure an app.

Mini Quiz

1. What do app configuration policies do?
2. How do they differ from app protection?
3. Name the two delivery types.
4. Where do the keys/values come from?
5. What must the app do to use them?

Answers: 1) Pre-configure app settings. 2) Configuration sets up the app; protection secures its data. 3) Managed devices (MDM) and Managed apps (MAM). 4) The app vendor's documentation. 5) Support app configuration.

7. Application Updates

Introduction

Application updates keep deployed software current — for security patches, bug fixes, and new features. How you update depends on the app type: Microsoft Store apps update themselves, Microsoft 365 Apps follow their update channel, and Win32/LOB apps are updated by you using **supersedence** (deploying a new version that replaces the old one). Keeping apps patched is a key part of reducing vulnerabilities.

Why This Topic Is Important

Outdated apps are a common vulnerability and support burden. Knowing the update mechanism for each app type — especially Win32 supersedence — is essential operational knowledge and an exam topic.

Key Concepts

- **Store app updates:** Automatic via the Microsoft Store (winget).
- **Microsoft 365 Apps updates:** Governed by the chosen **update channel** (Current/Monthly Enterprise/ Semi-Annual).
- **Win32/LOB supersedence:** Deploy a newer app version configured to **supersede** (and optionally uninstall) the previous one.
- **Supersedence chains:** Sequential version replacements; keep chains short and clear.
- **Detection rules & versioning:** Updated detection rules confirm the new version installed.
- **Phased updates:** Pilot a new app version before fleet-wide rollout.

Update method by app type

App type	Update method	Admin effort
Microsoft Store app (new)	Automatic via Store/winget	None
Microsoft 365 Apps	Update channel (Click-to-Run)	Choose/govern channel
Win32 / LOB	Supersedence (new version replaces old)	Package + configure supersedence

Architecture / Workflow (Win32 supersedence)

1. Package the **new version** as a Win32 app with updated detection rules.
2. Configure **supersedence**, referencing the old app and choosing to update or uninstall-then-install.
3. Assign the new version; Intune replaces the old one on targeted devices.
4. Detection confirms the new version; status is reported.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → (Win32 app) → Properties → Supersedence
```

```
Intune Admin Center → Apps → (Microsoft 365 Apps) → update channel configuration
```

```
Intune Admin Center → Apps → Monitor → App install status
```

Step-by-Step Configuration

A. Update a Win32 app via supersedence

1. Add the **new version** as a Win32 app (with correct detection for the new version).
2. Open *Supersedence* and add the **previous app** as superseded.
3. Choose update behavior (replace, or uninstall previous first).
4. Assign and monitor; old version is replaced.

B. Govern Microsoft 365 Apps updates

1. Set the desired **update channel** in the Microsoft 365 Apps config.
2. Optionally pin/manage Office updates via policy.

Real-World Scenario

Scenario: LOB app upgrade

The internal HR app releases v2 with security fixes. The admin packages v2, configures **supersedence** over v1 (uninstall v1, install v2), pilots it on a test group, then rolls it out fleet-wide — every device cleanly upgrades with no user action.

Enterprise Deployment Example

An enterprise lets the Store and Office channels auto-update mainstream apps, while a packaging team manages Win32/LOB updates through supersedence chains with pilot-then-broad rollout. App install reports confirm version adoption, and rollback is possible by deploying the prior packaged version if a new release misbehaves.

Troubleshooting

Old app version not replaced

Symptoms: Both versions present, or old version remains.

Root Cause: Supersedence misconfigured, detection rules overlap, or assignment gap.

Steps: Verify supersedence references the correct prior app; ensure new-version detection is accurate; confirm assignment; review install status.

Resolution: Fix supersedence/detection; the new version replaces the old.

Microsoft 365 Apps on wrong version

Symptoms: Office updates too fast/slow.

Root Cause: Incorrect or conflicting update channel.

Steps: Confirm the configured channel; remove conflicting update policies.

Resolution: Set the intended channel.

Best Practices

- Let **Store** and **Office channels** auto-update where possible.
- Use **supersedence** for Win32/LOB upgrades; keep chains short.
- Update **detection rules** for each new version.
- **Pilot** new app versions before broad rollout.
- Keep prior packages available for **rollback**.

Security Considerations

App updates close security vulnerabilities, so timely updates are a security control, not just maintenance. Automate where you can (Store/Office) and operationalize Win32 updates with pilots and monitoring. Validate update packages from trusted sources, since the update channel is also a potential attack vector if misused.

PowerShell Commands

```
# Check available Store/winget updates locally
winget upgrade

# Office version/channel
Get-ItemProperty "HKLM:\SOFTWARE\Microsoft\Office\ClickToRun\Configuration" |
  Select-Object VersionToReport, UpdateChannel

# Inspect app/supersedence relationships via Graph
Connect-MgGraph -Scopes "DeviceManagementApps.Read.All"
Get-MgDeviceAppManagementMobileApp | Select-Object DisplayName, Id
```

Interview Questions and Answers

BEGINNER **How do Microsoft Store apps update?**

Automatically via the Store (winget), with no admin effort.

BEGINNER **How do Win32 apps update?**

You package a new version and use supersedence to replace the old one.

INTERMEDIATE **What controls Microsoft 365 Apps updates?**

The update channel (Current, Monthly Enterprise, or Semi-Annual Enterprise) under Click-to-Run.

INTERMEDIATE **Why update detection rules with each Win32 version?**

So Intune correctly detects the new version as installed; stale rules cause wrong status or repeated installs.

ADVANCED **A superseded old version remains — what's wrong?**

Supersedence is misconfigured or detection overlaps; verify the supersedence reference, new-version detection, and assignment.

MD-102 Exam Notes

Exam Focus

Update method by type: **Store = automatic**, **Microsoft 365 Apps = update channel**, **Win32/LOB = supersedence**. Update **detection rules** per version. Pilot before broad rollout. Keep prior packages for rollback.

Hands-On Lab

Lab 7.1 — Update apps

1. Package a "v2" of a Win32 app with updated detection. 2. Configure supersedence over the v1 app. 3. Assign and confirm v1 is replaced by v2. 4. Change the Microsoft 365 Apps update channel and verify. 5. Run `winget upgrade` to view Store updates.

Summary

Application updates vary by type: Store apps update automatically, Microsoft 365 Apps follow their update channel, and Win32/LOB apps use supersedence to replace older versions. Update detection rules per version, pilot before broad rollout, and keep prior packages for rollback — treating updates as a security control.

Mini Quiz

1. How do Store apps update?
2. What controls Office updates?
3. What mechanism updates Win32 apps?
4. What must change with each Win32 version?
5. What enables rollback of a bad update?

Answers: 1) Automatically via the Store/winget. 2) The update channel. 3) Supersedence. 4) Detection rules. 5) Keeping the prior packaged version available.

8. Application Troubleshooting

Introduction

Application troubleshooting is the skill of diagnosing why an app didn't install, won't update, or reports the wrong status. Most app problems trace to a small set of causes — detection rules, assignments, requirements, dependencies, conflicts, or connectivity — and the **Intune Management Extension (IME) log** plus **App install status** are your primary tools. This topic consolidates a repeatable diagnostic method.

Why This Topic Is Important

App failures generate help-desk tickets and block productivity. A systematic troubleshooting approach is essential daily work and a frequent interview theme ("walk me through troubleshooting a failed app install"). The exam expects you to know where the logs and status live.

Key Concepts

- **App install status:** Per-device/user status with error codes (Apps → Monitor).
- **IME log:** `IntuneManagementExtension.log` on the device — the main Win32/script log.
- **Detection rule failures:** The most common cause of wrong status/repeated installs.
- **Requirement/dependency failures:** App skipped because conditions/prerequisites aren't met.
- **Conflicts:** Another app/policy interfering (e.g., conflicting Office).
- **Connectivity:** Device can't reach the content CDN/Store.
- **Sync timing:** Apps arrive at check-in, not instantly.
- **Error codes:** Map exit codes to causes (e.g., MSI 1603 = fatal install error).

Diagnostic Workflow

1. Open **App install status** and note the **error code** and stage.
2. Check the **detection rule** — does it match what the installer creates?
3. Verify **requirements** (OS, architecture, disk) and **dependencies**.
4. Look for **conflicts** (existing app/version, conflicting policy).
5. Confirm **connectivity** and that the device has **synced**.
6. Read the **IME log** on the device for the detailed failure.
7. Test the **install command** locally to validate switches.

Microsoft Portal Navigation

```
Intune Admin Center → Apps → Monitor → App install status
```

```
Intune Admin Center → Apps → All apps → (app) → Device install status / User install status
```

```
On device: C:
\ProgramData\Microsoft\IntuneManagementExtension\Logs\IntuneManagementExtension.log
```

Common Failures (Symptoms → Root Cause → Resolution)

Win32 app install fails (error 1603 or similar)

Symptoms: Failed status, MSI/exit error code.

Root Cause: Bad install switches, unmet requirement, missing dependency, or conflicting prior install.

Steps: Test the silent command locally; verify requirements/dependencies; remove conflicts; read the IME log.

Resolution: Correct command/requirements; redeploy.

App keeps reinstalling / wrong status

Symptoms: Repeated install attempts; "Failed" though installed.

Root Cause: Detection rule doesn't match the installed state.

Steps: Correct the detection rule (path/version/registry/MSI code).

Resolution: Accurate detection; status stabilizes.

App not offered at all

Symptoms: App never appears/installs.

Root Cause: Not assigned, requirement excludes the device, or not synced/licensed.

Steps: Confirm assignment and requirements; verify license; force sync.

Resolution: Fix assignment/requirements; app is delivered.

App protection/config not enforced

Symptoms: No PIN or unconfigured app.

Root Cause: User not signed in with work account, policy unassigned, app unsupported, or wrong delivery type.

Steps: Confirm work-account sign-in, assignment, app support, and delivery type.

Resolution: Correct sign-in/assignment; policy enforces on next launch.

Real-World Scenario

> **Scenario: The phantom failure**
>
> A Win32 app shows "Failed" on hundreds of devices, yet users confirm it's installed. The admin checks App install status, sees a detection mismatch, and discovers the detection rule pointed to an old file version. Updating the detection rule to the correct version flips everything to "Installed" without reinstalling anything.

Enterprise Deployment Example

An enterprise runs an app-health process: App install status is reviewed daily, failures are triaged by error code, IME logs are collected for stubborn cases, and a knowledge base maps common exit codes to fixes. Detection-rule standards and local install-command testing are mandated before any Win32 app ships, sharply reducing failures.

Best Practices

- Start with **App install status** and the **error code**.
- Treat **detection rules** as the first suspect for status problems.
- Always **test install commands locally** before packaging.
- Read the **IME log** for Win32/script detail.
- Check **assignment, requirements, dependencies, conflicts, connectivity, sync** systematically.

Security Considerations

Failed security-relevant apps (an EDR agent, a VPN) are a risk, so prioritize their remediation. Be cautious sharing IME logs (they can contain environment detail), and ensure only trusted packages are redeployed during troubleshooting. A repeatable, documented process reduces both downtime and the chance of introducing insecure workarounds.

PowerShell Commands

```
# Read the Intune Management Extension log (Win32/scripts)
Get-Content "C:
\ProgramData\Microsoft\IntuneManagementExtension\Logs\IntuneManagementExtension.log" -
Tail 100

# Confirm an MSI product is installed (for detection validation)
Get-WmiObject Win32_Product | Where-Object { $_.Name -like "*AppName*" }

# Force an Intune sync from the device
Get-ScheduledTask | ? {$_.TaskName -like "*PushLaunch*"} | Start-ScheduledTask
```

Interview Questions and Answers

BEGINNER **Where do you first look when an app fails to install?**

Apps → Monitor → App install status for the error code and stage.

BEGINNER **What's the main on-device log for Win32 apps?**

IntuneManagementExtension.log under C:\ProgramData\Microsoft\IntuneManagementExtension\Logs.

INTERMEDIATE **An app reinstalls repeatedly — likely cause?**

A detection rule that doesn't match the installed state, so Intune thinks it's not installed.

INTERMEDIATE **An app is never offered to a device — what do you check?**

Assignment, requirement rules, licensing, and whether the device has synced.

ADVANCED **Walk through troubleshooting a Win32 failure with error 1603.**

Check App install status for context, test the silent install command locally, verify requirements/dependencies and return codes, look for conflicting installs, and read the IME log for the detailed cause; then fix and redeploy.

MD-102 Exam Notes

Exam Focus

Primary tools: **App install status** and the **IntuneManagementExtension.log**. **Detection rules** cause most status issues; **requirements/dependencies/conflicts/connectivity/sync** cause delivery issues. Test install commands locally. Know common codes (e.g., MSI **1603** = fatal install error).

Hands-On Lab

Lab 8.1 — Diagnose an app failure

1. Deliberately misconfigure a Win32 app's detection rule and deploy it. 2. Observe the "Failed" status in App install status. 3. On the device, read the IME log. 4. Correct the detection rule and resync. 5. Confirm the status flips to Installed. Document the steps as a runbook.

Summary

Application troubleshooting follows a repeatable path: read App install status and the error code, suspect detection rules first, verify requirements/dependencies/conflicts/connectivity/sync, and dig into the IME log for Win32 detail. Testing install commands locally and maintaining a documented process keeps app delivery reliable and secure.

Mini Quiz

1. Where do you see app error codes?
2. What on-device log covers Win32 installs?
3. What's the first suspect for wrong install status?
4. Name two causes of an app never being offered.
5. What does MSI error 1603 indicate?

Answers: 1) Apps → Monitor → App install status. 2) IntuneManagementExtension.log. 3) The detection rule. 4) Missing assignment, unmet requirement (also licensing or no sync). 5) A fatal install error.

9. Section 4 — Consolidated Interview Preparation

Introduction

Application management questions test whether you can deliver, secure, and maintain software at scale. Below are the most likely questions grouped as employers ask them. Answer each in two or three sentences: concept, Microsoft feature, exact tool or portal path.

Most Asked Interview Questions

1. What are the app assignment intents?

Required (auto-install), Available for enrolled devices (Company Portal self-service), and Uninstall.

2. Store apps vs Win32 apps?

Store apps need no packaging and auto-update (mainstream software); Win32 apps are packaged into .intunewin for full control over complex/legacy apps.

3. How do you deploy Office?

Use the Microsoft 365 Apps app type (Click-to-Run), choosing apps, 64-bit architecture, and an update channel like Monthly Enterprise.

4. What is an app protection policy?

A MAM policy that secures corporate data inside apps (PIN, encryption, copy/paste limits, selective wipe) without managing the device — ideal for BYOD.

5. App protection vs app configuration?

Protection secures app data; configuration pre-sets app settings (server URL, account) so the app works on first launch.

Scenario-Based Questions

1. Deploy a custom LOB app with a prerequisite runtime.

Package it as Win32 (.intunewin), set the runtime as a dependency, define a detection rule, and assign as Required.

2. Secure corporate email on personal phones.

Apply an app protection policy to Outlook (PIN, encryption, restricted copy/paste) and require an app protection policy via Conditional Access.

3. Upgrade an internal app to v2 cleanly.

Package v2 with updated detection and configure supersedence over v1 (uninstall v1, install v2); pilot then roll out.

4. Pre-configure an app so users don't set it up.

Create an app configuration policy (Managed devices or Managed apps) with the vendor's keys/values and assign it.

Troubleshooting Questions

1. A Win32 app reports Failed but is installed. Cause?

A detection rule that doesn't match the installed state; correct the path/version/registry/MSI code.

2. Which log do you read for a Win32 install failure?

IntuneManagementExtension.log on the device.

3. Office won't install on some devices. Common cause?

A conflicting existing Office install or architecture mismatch; remove the old MSI Office first.

Practical Administrator Questions

1. How do you offer optional apps for self-service?

Assign them as Available for enrolled devices so users install from Company Portal.

2. How do you remove an app fleet-wide?

Assign the app with the Uninstall intent to the target group.

3. How do you offboard a BYOD user's corporate data?

Issue a selective wipe via app protection, removing only corporate app data.

Production Environment Questions

1. Design an app strategy for a mixed fleet.

Store apps for mainstream software, Microsoft 365 Apps for Office, Win32 for custom/legacy apps, app protection for BYOD, app configuration to pre-set apps, and supersedence for updates — monitored via App install status.

2. How do you keep apps patched?

Let Store/Office auto-update; manage Win32/LOB updates via supersedence with pilot-then-broad rollout.

3. How do you reduce app-related tickets?

Pre-configure apps with configuration policies, get detection rules right, offer self-service via Company Portal, and monitor install status proactively.

Summary

Expect deep questions on assignment intents, Store vs Win32, Office deployment, app protection (MAM) vs app configuration, updates via supersedence, and systematic troubleshooting. Answer with the business goal, the specific feature, and the exact portal path or tool.

10. Section 4 — Troubleshooting Reference

Introduction

A symptom-first reference for the application issues you'll meet most. Each entry follows Symptoms → Root Cause → Steps → Resolution.

1. App not installing (Required)

Symptoms: Failed/Pending; app absent.

Root Cause: Wrong detection rule, unmet requirement, missing dependency, or no sync.

Steps: Read App install status error code; verify detection/requirements/dependencies; force sync.

Resolution: Correct config; app installs on next check-in.

2. App reports wrong status / reinstalls

Symptoms: "Failed" though installed, or repeated installs.

Root Cause: Detection rule doesn't match installed state.

Steps: Fix the detection rule (path/version/registry/MSI code).

Resolution: Accurate detection; status corrects.

3. Win32 install failure (e.g., 1603)

Symptoms: Failed with an exit code.

Root Cause: Bad install switches, conflict, or unmet requirement.

Steps: Test the silent command locally; read the IME log; verify requirements/return codes.

Resolution: Fix command/requirements; redeploy.

4. Store app won't install

Symptoms: Store app fails.

Root Cause: Store/winget blocked, offline, or platform/region unsupported.

Steps: Verify Store connectivity and assignment; force sync.

Resolution: Restore access; app installs.

5. Microsoft 365 Apps fails/conflicts

Symptoms: Office won't install or wrong version.

Root Cause: Conflicting Office/architecture or wrong update channel.

Steps: Remove old MSI Office/match bitness; confirm channel; check assignment.

Resolution: Clean conflicts; set channel.

6. App protection not enforced

Symptoms: No PIN/data controls.

Root Cause: Policy unassigned, app not managed, no work-account sign-in, or no license.

Steps: Confirm assignment, managed-app list, work-account sign-in, licensing.

Resolution: Correct sign-in/assignment; enforces on next launch.

7. App configuration ignored

Symptoms: App opens unconfigured.

Root Cause: Wrong delivery type, unsupported app, bad keys, or unassigned.

Steps: Verify delivery type/app support/keys/assignment; sync.

Resolution: Correct config; app reads it on next launch.

8. Old version not replaced (supersedence)

Symptoms: Both versions present.

Root Cause: Supersedence misconfigured or detection overlap.

Steps: Verify supersedence reference and new-version detection; confirm assignment.

Resolution: Fix supersedence/detection; new version replaces old.

Best Practices

- Start with **App install status** + error code, then the **IME log**.
- Suspect **detection rules** first for status problems.
- Check **assignment, requirements, dependencies, conflicts, connectivity, sync**.
- Test **install commands locally** before packaging.
- Use **selective wipe** for BYOD offboarding; **supersedence** for clean upgrades.

Summary

Most Section 4 issues trace to detection rules, assignment/requirement gaps, conflicts (especially Office), connectivity, sync timing, or supersedence misconfiguration. App install status and the IntuneManagementExtension.log, combined with local command testing, resolve the majority quickly.

End of Section 4 — Manage Applications.
This completes all four MD-102 curriculum domains. Next is the *Final Reference Appendix* (exam strategy, practical lab checklist, daily tasks, Intune & Entra cheat sheets, common exam mistakes, last-minute revision, mock interview questions, enterprise best practices, the endpoint administrator checklist, the PowerShell cheat sheet, and the 50 most asked Intune interview questions), after which all parts will be merged into a single continuous master PDF.

MICROSOFT CERTIFICATION

MD-102

Microsoft Endpoint Administrator

Final Reference Appendix

Exam Strategy · Lab Checklist · Daily Tasks · Portal Cheat Sheets
Common Mistakes · Last-Minute Revision · Mock Interviews · Best Practices
Endpoint Administrator Checklist · PowerShell Cheat Sheet · 50 Interview Questions

The complete quick-reference companion to Sections 1–4

Appendix Contents

1. MD-102 Exam Preparation Strategy
2. Practical Lab Checklist
3. Intune Administrator Daily Tasks
4. Intune Portal Cheat Sheet
5. Microsoft Entra Portal Cheat Sheet
6. Common MD-102 Exam Mistakes
7. Last-Minute Revision Notes
8. Mock Interview Questions
9. Enterprise Best Practices
10. Real-World Endpoint Administrator Checklist
11. PowerShell Cheat Sheet for Intune Administrators
12. 50 Most Asked Intune Interview Questions

How to use this appendix: This is your fast-access companion to the four study sections. Use the cheat sheets the week before the exam, the checklists on the job, and the 50 interview questions for final preparation. Everything here distills and reinforces the detailed material in Sections 1–4.

1. MD-102 Exam Preparation Strategy

About the Exam

MD-102 (Microsoft Endpoint Administrator) validates your ability to deploy, configure, protect, manage, and monitor devices and client applications in a Microsoft 365 environment. It is the single exam for the Endpoint Administrator Associate certification. Expect a mix of multiple-choice, multiple-response, drag-and-drop ordering, and case-study questions.

Skills Measured (weightings are approximate and change over time)

Domain	Maps to	Approx. weight
Deploy Windows client	Section 1	~25%
Manage identity and compliance	Section 2	~25%
Manage, maintain, and protect devices	Section 3	~30%
Manage applications	Section 4	~20%

> Always confirm the current weightings and objectives on the official Microsoft Learn MD-102 page before your exam, as Microsoft updates them periodically.

A 4-Week Study Plan

Week	Focus	Activities
1	Deploy + Identity (Sections 1–2)	Read sections; build Autopilot, Entra Join, compliance, and Conditional Access labs in a trial tenant.
2	Manage/Protect (Section 3)	Labs for enrollment, configuration profiles, endpoint security, Defender for Endpoint, remote actions, updates.
3	Applications (Section 4)	Package a Win32 app, deploy Store + M365 Apps, app protection/ configuration, supersedence.
4	Review + practice	Cheat sheets, common mistakes, last-minute notes; take practice tests; redo weak labs.

Study Tactics That Work

- **Hands-on first:** Build everything in a free Microsoft 365 developer/trial tenant — the exam rewards practical familiarity with portal paths.
- **Learn the "which tool" decisions:** Autopilot vs provisioning package, Entra Join vs Hybrid, MDM vs MAM, Store vs Win32, compliance vs Conditional Access.
- **Memorize portal navigation** for high-frequency tasks (this appendix's cheat sheets).
- **Master the comparison tables** — they map directly to exam distractors.

- **Practice case studies** under time pressure; read the requirements carefully before the questions.

Exam-Day Tips

- Read each scenario's **requirements and constraints** first; they dictate the "best" answer.
- Prefer **cloud-native** answers unless on-prem dependencies are stated.
- Flag and return to long case studies; don't let one item consume your time budget.
- Watch for "least administrative effort" and "most secure" qualifiers — they change the correct choice.
- Eliminate obviously wrong options first to improve your odds on tough items.

2. Practical Lab Checklist

Complete these labs in a trial tenant. If you can do every one from memory, you're exam- and job-ready.

Deploy (Section 1)

- ☐ Register a device in Autopilot (hardware hash) and create a User-Driven deployment profile.
- ☐ Configure the Enrollment Status Page to block until apps install.
- ☐ Build a provisioning package in Windows Configuration Designer with a bulk Entra token.
- ☐ Entra-join a device and confirm automatic Intune enrollment.
- ☐ Verify join state with `dsregcmd /status`.
- ☐ Create Pilot and Broad Windows Update rings.

Identity & Compliance (Section 2)

- ☐ Create dynamic user and device groups.
- ☐ Build a Windows compliance policy (BitLocker, min OS) with a grace period.
- ☐ Set the tenant default to "Not compliant."
- ☐ Create Conditional Access policies (require MFA, require compliant device) in report-only.
- ☐ Create a break-glass account and exclude it from CA.
- ☐ Enable Authenticator with number matching; enable SSPR.
- ☐ Deploy a BitLocker disk encryption policy with Entra key escrow.
- ☐ Retrieve a BitLocker recovery key from Entra.

Manage & Protect (Section 3)

- ☐ Configure Windows automatic enrollment and an enrollment restriction.
- ☐ Create a Settings Catalog configuration profile and a Wi-Fi profile.
- ☐ Create an Administrative Templates profile (Edge/Office).
- ☐ Apply a device restriction profile and a single-app kiosk.
- ☐ Deploy the Windows security baseline to a pilot group.
- ☐ Create an ASR policy (Audit) and an antivirus policy.
- ☐ Connect Defender for Endpoint and onboard a device via EDR policy.
- ☐ Add a machine-risk rule to compliance.
- ☐ Issue remote actions (Sync, Retire) and review results.
- ☐ Enable Endpoint analytics and create a proactive remediation.

Applications (Section 4)

- ☐ Deploy a Required app and an Available app (Company Portal).
- ☐ Add a Microsoft Store app (new).

- ☐ Package a Win32 app (.intunewin) with detection rules and deploy it.
- ☐ Deploy Microsoft 365 Apps (64-bit, Monthly Enterprise Channel).
- ☐ Create an app protection policy and perform a selective wipe.
- ☐ Create an app configuration policy.
- ☐ Update a Win32 app via supersedence.
- ☐ Diagnose a failed install using App install status and the IME log.

3. Intune Administrator Daily Tasks

A realistic picture of the role — useful for interviews ("what does your day look like?") and for operating a real tenant.

Daily

- Review the **device compliance** dashboard and address newly non-compliant devices.
- Check **app install status** for failures and remediate.
- Triage **Defender for Endpoint** alerts / high-risk devices.
- Process **enrollment** issues and new-device onboarding.
- Handle help-desk escalations (BitLocker recovery, MFA resets, remote actions).

Weekly

- Review **Conditional Access** sign-in logs and report-only insights.
- Validate **Windows Update** ring progress and pending feature updates.
- Review **Endpoint analytics** (startup performance, app reliability).
- Audit **configuration profile** conflicts and failures.
- Check **Apple token** and certificate expiry dates.

Monthly / Periodic

- Review and update **security baselines** to current versions.
- Clean up **stale device objects** in Entra.
- Pilot and roll out **app updates** (supersedence) and new app versions.
- Review **RBAC** assignments and scope tags.
- Validate **break-glass accounts** and Secure Score improvements.
- Produce **compliance/encryption reports** for stakeholders.

4. Intune Portal Cheat Sheet

High-frequency navigation paths in the Microsoft Intune Admin Center (`intune.microsoft.com`). Memorize these for the exam and daily work.

Devices & Enrollment

Task	Path
All devices	Devices → All devices
Windows automatic enrollment	Devices → Enrollment → Windows → Automatic Enrollment
Autopilot devices & profiles	Devices → Enrollment → Windows → Windows enrollment
Enrollment Status Page	Devices → Enrollment → Windows → Enrollment Status Page
Apple enrollment (ADE/ABM)	Devices → Enrollment → Apple
Android enrollment	Devices → Enrollment → Android
Enrollment restrictions	Devices → Enrollment → Enrollment restrictions
Corporate device identifiers	Devices → Enrollment → Corporate device identifiers

Configuration & Compliance

Task	Path
Configuration profiles (Settings Catalog/ Templates)	Devices → Configuration
Administrative Templates	Devices → Configuration → Create → Templates → Administrative Templates
Compliance policies	Devices → Compliance → Policies
Compliance default ("no policy = ...")	Devices → Compliance → Settings
Device compliance monitoring	Devices → Monitor → Device compliance
Encryption report	Devices → Monitor → Encryption report

Endpoint Security

Task	Path
Security baselines	Endpoint security → Security baselines
Antivirus / Firewall	Endpoint security → Antivirus / Firewall
Disk encryption (BitLocker)	Endpoint security → Disk encryption
EDR onboarding	Endpoint security → Endpoint detection and response
Attack surface reduction	Endpoint security → Attack surface reduction
Defender for Endpoint connector	Endpoint security → Microsoft Defender for Endpoint

Apps

Task	Path
Add/assign apps	Apps → All apps → Add
App install status	Apps → Monitor → App install status
App protection policies	Apps → App protection policies
App configuration policies	Apps → App configuration policies

Updates, Reports & Admin

Task	Path
Update rings / Feature updates / Expedite	Devices → Windows → Windows updates
Endpoint analytics	Reports → Endpoint analytics
Windows update reports	Reports → Windows updates
RBAC roles & scope tags	Tenant administration → Roles
Proactive remediations	Devices → Windows → Proactive remediations

5. Microsoft Entra Portal Cheat Sheet

High-frequency navigation in the Microsoft Entra Admin Center (`entra.microsoft.com`).

Identity

Task	Path
Users	Identity → Users → All users
Groups (incl. dynamic)	Identity → Groups → All groups
Roles & administrators	Identity → Roles & admins
Devices (all / settings)	Identity → Devices → All devices / Device settings
BitLocker keys (per device)	Identity → Devices → (device) → BitLocker keys
Hybrid join (Entra Connect)	Identity → Hybrid management → Microsoft Entra Connect

Protection

Task	Path
Conditional Access policies	Protection → Conditional Access → Policies
Named locations	Protection → Conditional Access → Named locations
CA report-only insights	Protection → Conditional Access → Insights and reporting
Authentication methods	Protection → Authentication methods → Policies
Authentication strengths	Protection → Authentication methods → Authentication strengths
Self-service password reset (SSPR)	Protection → Password reset
Identity Protection (risk, P2)	Protection → Identity Protection

Monitoring

Task	Path
Sign-in logs (CA outcomes)	Monitoring & health → Sign-in logs
Audit logs	Monitoring & health → Audit logs

6. Common MD-102 Exam Mistakes

Avoid these frequent traps that cost candidates marks.

Conceptual Confusions

- **Compliance policy ≠ enforcement.** A compliance policy only *labels* a device; Conditional Access does the blocking. A question asking "how do you *block* non-compliant devices?" wants CA, not the compliance policy alone.
- **Registration vs enrollment.** Registration/join = identity in Entra; enrollment = management in Intune. Don't conflate them.
- **Entra registered vs joined vs hybrid.** Registered = BYOD (user-owned); Joined = corporate cloud-only; Hybrid = on-prem AD + Entra. Pick based on stated dependencies.
- **Wipe vs Retire.** Wipe = whole device; Retire = corporate data only (BYOD). Choosing Wipe for a personal device is wrong (data-loss/privacy).
- **MDM vs MAM.** MDM = whole device (corporate); MAM/app protection = data in apps (BYOD).
- **Store vs Win32.** Store for mainstream/auto-update; Win32 for complex/legacy/custom.
- **Configuration vs protection (apps).** Configuration sets up the app; protection secures its data.

Process & Detail Mistakes

- Forgetting **Self-deploying Autopilot requires TPM 2.0**.
- Forgetting that **provisioning-package bulk tokens expire**.
- Forgetting **BitLocker key escrow** must be configured before silent encryption.
- Ignoring **report-only mode** and **break-glass accounts** when deploying CA.
- Assuming a **configuration profile** works without enrollment (it needs MDM enrollment, unlike GPO needing domain join).
- Overlooking that **detection rules** drive Win32 install status.
- Not removing **conflicting/old Office** before deploying Microsoft 365 Apps.
- Choosing on-prem/hybrid answers when the scenario is clearly **cloud-native**.

Question-Reading Mistakes

- Missing qualifiers like **"least administrative effort"** or **"most secure."**
- Not noticing **licensing constraints** (e.g., CA needs Entra ID P1; risk-based needs P2).
- Answering before reading **all requirements** in a case study.

7. Last-Minute Revision Notes

Read these the night before and the morning of the exam. Each line is a high-yield fact.

Deploy

- Autopilot modes: **User-Driven, Self-deploying (TPM 2.0), Pre-provisioning, Existing devices**.
- Autopilot targets devices via a **dynamic group (ZTDId rule)**; ESP can **block** until apps install.
- Provisioning packages = **.ppkg** from **Windows Configuration Designer**; bulk Entra token **expires**.
- Subscription Activation: Pro → Enterprise via the user's **license**; needs **Entra join**; steps down when removed.
- Entra Join = cloud-only; **Hybrid** needs **Entra Connect + SCP**; both YES in `dsregcmd /status`.
- WUfB sends **policy, not content**; rings/deferrals; **Expedite** for emergencies; **Autopatch** automates rings.

Identity & Compliance

- Entra ID **P1 = Conditional Access**; **P2 = Identity Protection + PIM**.
- Group membership: **Assigned** (manual) vs **Dynamic** (rule).
- Compliance *labels*; **Conditional Access enforces**. Set tenant default to **Not compliant**.
- CA = **assignments (if) + access controls (then)**; use **report-only**; keep **break-glass** excluded; **block legacy auth**.
- **Number matching** defeats MFA fatigue; **FIDO2/Windows Hello** are phishing-resistant.
- BitLocker: **full disk**, **48-digit key** escrowed to Entra, **silent enablement** needs TPM. EFS = individual files.
- Troubleshoot access in **Entra sign-in logs (CA tab)**; compliance via **per-setting status**.

Manage & Protect

- **MDM** = whole device; **MAM** = app data. RBAC + **scope tags** for delegation.
- Enrollment: Windows **Autopilot/auto-enroll**; Apple **ADE/ABM**; Android **fully managed / work profile**; **Apple tokens expire**.
- **Settings Catalog** = modern profile; **Administrative Templates** = ADMX/GPO-like; **OMA-URI** = custom. Conflicts **not applied**.
- Endpoint security: **baselines**, AV, firewall, disk encryption, **ASR (Audit→Block)**, **EDR**, account protection.
- MDE = **EDR**; flow: **onboard → risk score → compliance (machine risk) → CA**. Defender AV = prevention.
- Remote actions: **Wipe** (all), **Retire** (corp data), **Fresh Start** (keep user data). Run at **next contact** if offline.
- Monitoring: **Endpoint analytics** + **proactive remediations** (detection + remediation scripts).

Applications

- Intents: **Required** (auto), **Available** (Company Portal), **Uninstall**.
- **Store app (new)** = no packaging, auto-update, winget-backed; **Win32** = .intunewin via **Content Prep Tool**.
- Win32 needs **detection rules** (top failure cause), requirements, dependencies, return codes, supersedence.
- Office = **Microsoft 365 Apps** (Click-to-Run); channels: Current / **Monthly Enterprise** / Semi-Annual; 64-bit default.
- **App protection (MAM)**: PIN, encryption, copy/paste limits, **selective wipe**, conditional launch; pair with CA.
- **App configuration** ≠ protection: pre-sets app settings; Managed devices (MDM) or Managed apps (MAM).
- Troubleshoot apps with **App install status** + **IntuneManagementExtension.log**; MSI **1603** = fatal install error.

8. Mock Interview Questions

Practice these aloud as a timed mock interview. Each should take 60–90 seconds to answer well.

Round 1 — Fundamentals (rapid fire)

1. What is Microsoft Intune and what problem does it solve?
2. Explain Entra registered vs joined vs hybrid joined.
3. What's the difference between MDM and MAM?
4. Compliance policy vs Conditional Access — how do they work together?
5. Store apps vs Win32 apps — when do you use each?

Round 2 — Scenario design

1. Design zero-touch deployment for a fully remote company.
2. Secure corporate email on employees' personal phones.
3. Ensure only encrypted, compliant devices reach SharePoint.
4. Automatically contain a compromised device.
5. Patch 10,000 devices safely without breaking the business.

Round 3 — Troubleshooting

1. An Autopilot device shows generic OOBE — diagnose it.
2. `dsregcmd` shows DomainJoined=YES, AzureAdJoined=NO — what's wrong?
3. A Win32 app reports Failed but is installed — why?
4. A user is blocked and you suspect Conditional Access — where do you look?
5. A BitLocker recovery key isn't in Entra — what happened?

Round 4 — Production & judgment

1. How would you migrate 20,000 domain-joined PCs to the cloud?
2. How do you implement least-privilege administration in Intune?
3. How do you prove device encryption and compliance to an auditor?
4. What's your approach to rolling out a new security baseline safely?
5. Walk me through your typical day as an endpoint administrator.

> **How to answer well**
>
> Structure each answer as: (1) the business goal, (2) the specific Microsoft feature, (3) the exact tool/portal path, and (4) one risk or best practice. This signals both conceptual and hands-on competence.

9. Enterprise Best Practices

Identity & Access

- Make **identity the perimeter**: enforce MFA everywhere, block legacy auth, use phishing-resistant methods for admins.
- Apply **least privilege** with scoped roles and **PIM** (just-in-time) for privileged access.
- Maintain two cloud-only **break-glass accounts** excluded from Conditional Access.
- Deploy Conditional Access in **report-only first**, then enforce in layers.

Device Management

- Default to **cloud-native** (Autopilot + Entra Join + Intune); use hybrid/co-management only as a bridge.
- Target policies with **dynamic groups**; keep user/device attributes clean.
- Use **RBAC + scope tags** for delegated administration.
- Standardize on a small set of **hardware models** to simplify support.

Security & Compliance

- Set the compliance tenant default to **Not compliant** (Zero Trust).
- Require **BitLocker** with **Entra key escrow**; enforce via compliance + CA.
- Deploy **security baselines**, **ASR (Audit→Block)**, and **Defender for Endpoint**; tie **risk to compliance**.
- Minimize antivirus **exclusions**; review Secure Score and TVM regularly.

Updates & Apps

- Patch with **phased rings** (Pilot → Broad); reserve **Expedite** for emergencies; consider **Autopatch**.
- Use **Store/Office channels** for auto-updates; **supersedence** for Win32; pilot before broad rollout.
- Secure BYOD with **app protection**; pre-configure apps with **app configuration**.
- Get **detection rules** right; monitor **App install status**.

Operations

- Operate from a **single console** (Intune Admin Center) for consistency.
- Build **runbooks** for onboarding, offboarding, lost/stolen devices, and incident response.
- Monitor **Endpoint analytics**; automate fixes with **proactive remediations**.
- Track **token/certificate expiry** (Apple, connectors) to avoid outages.
- Export to **Power BI/Graph** for leadership visibility and audit evidence.

10. Real-World Endpoint Administrator Checklist

A practical operating checklist for running a healthy Intune environment.

Tenant Foundation

- ☐ Intune set as MDM authority; automatic enrollment configured.
- ☐ Break-glass accounts created, excluded from CA, and securely stored.
- ☐ RBAC roles and scope tags defined for delegated administration.
- ☐ Dynamic groups established for users and devices.
- ☐ Apple/Android connectors and tokens configured with renewal reminders.

Security Posture

- ☐ MFA enforced; legacy authentication blocked.
- ☐ Conditional Access: require MFA + compliant device for M365.
- ☐ Compliance policies with Not-compliant default and grace periods.
- ☐ BitLocker enforced with Entra key escrow.
- ☐ Security baselines + ASR + Defender for Endpoint deployed.
- ☐ Machine risk wired into compliance and Conditional Access.

Deployment & Apps

- ☐ Autopilot profiles and Enrollment Status Page configured.
- ☐ Baseline Required apps assigned to all devices.
- ☐ Microsoft 365 Apps deployed (64-bit, Monthly Enterprise).
- ☐ App protection policies for BYOD; app configuration for managed apps.
- ☐ Win32 packaging standards and supersedence process documented.

Maintenance & Monitoring

- ☐ Windows Update rings (Pilot/Broad) operating; feature versions controlled.
- ☐ Endpoint analytics enabled; proactive remediations in place.
- ☐ Compliance/encryption reports reviewed on a cadence.
- ☐ Stale device cleanup scheduled.
- ☐ Runbooks for onboarding, offboarding, lost/stolen, and incidents.

11. PowerShell Cheat Sheet for Intune Administrators

Common commands consolidated from Sections 1–4. Connect to Microsoft Graph first, then use the relevant cmdlets.

Setup & Connection

```
# Install and import the Microsoft Graph SDK
Install-Module Microsoft.Graph -Scope CurrentUser -Force
Import-Module Microsoft.Graph

# Connect with the scopes you need
Connect-MgGraph -Scopes "DeviceManagementManagedDevices.Read.All",
  "DeviceManagementConfiguration.Read.All","DeviceManagementApps.Read.All",
  "Device.Read.All","Policy.Read.All"
```

Device Join & Local Diagnostics

```
dsregcmd /status                # Entra/Hybrid join, PRT, MDM enrollment
Get-Tpm                         # TPM presence/readiness (Hello/BitLocker)
Get-BitLockerVolume             # BitLocker status per volume
Get-MpComputerStatus            # Defender AV real-time/running mode
Get-Service -Name Sense         # Defender for Endpoint (EDR) sensor
Get-ComputerInfo | Select WindowsProductName, OsBuildNumber, WindowsVersion
```

Autopilot

```
Install-Script -Name Get-WindowsAutopilotInfo -Force
Get-WindowsAutopilotInfo -OutputFile C:\HWID\hash.csv   # capture hardware hash
Get-WindowsAutopilotInfo -Online                        # capture + upload
```

Devices & Compliance (Graph)

```
Get-MgDeviceManagementManagedDevice |
  Select DeviceName, ComplianceState, OSVersion, LastSyncDateTime
Get-MgDeviceManagementManagedDevice -All |
  Group-Object ComplianceState | Select Name, Count
Get-MgDevice -All | Select DisplayName, TrustType, IsManaged
```

Configuration, Apps & Security (Graph)

```
Get-MgDeviceManagementDeviceConfiguration | Select DisplayName
Get-MgDeviceManagementConfigurationPolicy | Select Name        # Settings Catalog
Get-MgDeviceManagementIntent | Select DisplayName              # baselines/intents
Get-MgDeviceAppManagementMobileApp | Select DisplayName, Id
Get-MgDeviceAppManagementiOSManagedAppProtection | Select DisplayName
```

Remote Actions (Graph)

```
Invoke-MgWipeDeviceManagementManagedDevice   -ManagedDeviceId <id>
Invoke-MgRetireDeviceManagementManagedDevice -ManagedDeviceId <id>
Sync-MgDeviceManagementManagedDevice         -ManagedDeviceId <id>
```

Win32 / App Troubleshooting (local)

```
# Package a Win32 app
IntuneWinAppUtil.exe -c C:\Source -s setup.exe -o C:\Output

# MSI silent install / uninstall
msiexec /i "App.msi" /qn
msiexec /x "{PRODUCT-GUID}" /qn

# Read the Intune Management Extension log
Get-Content "C:
\ProgramData\Microsoft\IntuneManagementExtension\Logs\IntuneManagementExtension.log" -
Tail 100
```

Updates

```
Get-HotFix | Sort-Object InstalledOn -Descending | Select -First 10
Install-Module PSWindowsUpdate -Force
Get-WindowsUpdate; Install-WindowsUpdate -AcceptAll -AutoReboot
winget upgrade                                  # Store/winget updates
```

12. 50 Most Asked Intune Interview Questions

The definitive question bank, with concise model answers. Cover the answer, attempt it aloud, then check yourself.

Fundamentals (1–10)

1. What is Microsoft Intune?

A cloud-based endpoint management service to enroll, configure, secure, and monitor devices and apps from one console, with no on-prem servers.

2. What is the difference between MDM and MAM?

MDM manages the whole device (corporate); MAM manages only corporate app data (BYOD), leaving personal data untouched.

3. What is Microsoft Entra ID?

Microsoft's cloud identity and access management service (formerly Azure AD) storing users, groups, and devices and controlling authentication and access.

4. Explain Entra registered, joined, and hybrid joined.

Registered = user-owned BYOD with a light identity; Joined = org-owned cloud-only device; Hybrid = device in both on-prem AD and Entra.

5. What is device enrollment?

Bringing a device under Intune management so it can receive policies and apps and report compliance.

6. What is the difference between registration and enrollment?

Registration/join gives the device an identity in Entra; enrollment puts it under Intune management.

7. What is a tenant?

An organization's dedicated, isolated instance of Entra ID / Microsoft 365.

8. Assigned vs dynamic group membership?

Assigned is maintained manually; dynamic is built automatically from a rule on user/device attributes.

9. Which Entra license unlocks Conditional Access?

Entra ID P1; P2 adds Identity Protection (risk-based policies) and PIM.

10. What is co-management?

Managing a device with both Configuration Manager and Intune simultaneously, typically as a cloud-migration bridge.

Deployment (11–18)

11. What is Windows Autopilot?

Cloud zero-touch deployment that turns a new OEM device into a managed corporate device on first sign-in, with no imaging.

12. Name the Autopilot deployment modes.

User-Driven, Self-deploying (needs TPM 2.0), Pre-provisioning, and Autopilot for existing devices.

13. What is a hardware hash?

A unique device fingerprint from firmware, used to register and identify the device in Autopilot.

14. What is the Enrollment Status Page?

A screen that blocks the desktop during provisioning while required apps, policies, and certificates install.

15. What is a provisioning package?

A .ppkg from Windows Configuration Designer that configures existing Windows devices offline (Wi-Fi, naming, bulk Entra join).

16. What is Subscription Activation?

Automatically stepping a device from Pro to Enterprise based on the signed-in user's license, with no keys; needs Entra join.

17. What does Microsoft Entra Connect do?

Synchronizes on-prem AD identities to Entra ID and enables Hybrid Entra Join (via the SCP).

18. Quality vs feature updates?

Quality updates are monthly security/reliability fixes; feature updates are major OS version upgrades.

Identity, Compliance & Security (19–32)

19. What is a device compliance policy?

Intune rules defining a healthy device (encryption, OS, PIN); Intune marks devices Compliant or Not compliant.

20. How do compliance policies and Conditional Access work together?

Compliance produces the status; Conditional Access enforces it (e.g., require a compliant device).

21. What is a grace period?

Time for a failing device to remediate before being marked noncompliant, preventing mass lockouts.

22. What is Conditional Access?

An Entra if-then policy engine that enforces access controls (MFA, compliant device, block) based on signals.

23. Name common grant controls.

Require MFA, require compliant device, require Entra-joined device, require app protection policy, or block access.

24. Why use report-only mode?

To see what a policy would do across real sign-ins without enforcing it, avoiding accidental lockouts.

25. What are break-glass accounts?

Emergency admin accounts excluded from Conditional Access to prevent total lockout.

26. Why block legacy authentication?

Legacy protocols can't perform MFA and are a common attacker bypass.

27. What is number matching?

An Authenticator feature requiring the user to type a number from the sign-in screen, defeating MFA-fatigue attacks.

28. Name a phishing-resistant authentication method.

FIDO2 security keys or Windows Hello for Business.

29. Why is a Windows Hello PIN safer than a password?

It's device-bound, never leaves the device, and only unlocks a TPM-protected key locally.

30. What does BitLocker do and where are keys stored?

Full-disk encryption; the 48-digit recovery key is escrowed to Entra ID.

31. BitLocker vs EFS?

BitLocker encrypts the whole disk (device theft); EFS encrypts individual files (other users on the system).

32. What is SSPR?

Self-service password reset using registered methods; hybrid needs password writeback.

Manage & Protect (33–42)

33. What is a configuration profile?

An Intune object pushing settings (Wi-Fi, VPN, certs, OS options) to enrolled devices — the cloud equivalent of Group Policy.

34. Configuration profiles vs Group Policy?

Profiles deliver settings from the cloud to any enrolled device; Group Policy needs domain join and DC line of sight.

35. What is the Settings Catalog?

The modern, searchable library of all configurable settings — the recommended profile type.

36. What happens with conflicting configuration settings?

The conflicting setting is not applied and is reported as a conflict.

37. What are security baselines?

Pre-configured sets of Microsoft-recommended security settings deployed as a hardened starting point.

38. What are ASR rules?

Attack surface reduction rules that block common attack techniques; pilot in Audit before Block.

39. Defender Antivirus vs Defender for Endpoint?

AV prevents/removes known malware; MDE adds EDR — behavioral detection, risk scoring, investigation, and response.

40. How does MDE integrate with Intune?

A connector feeds device risk into Intune compliance; Conditional Access then blocks devices over a risk threshold.

41. Wipe vs Retire vs Fresh Start?

Wipe factory-resets everything (corporate); Retire removes corporate data only (BYOD); Fresh Start reinstalls Windows keeping user data.

42. What are proactive remediations?

Script packages with detection and remediation scripts that automatically find and fix common device issues.

Applications (43–50)

43. Name the app assignment intents.

Required (auto-install), Available for enrolled devices (Company Portal), and Uninstall.

44. Store apps vs Win32 apps?

Store apps need no packaging and auto-update (mainstream); Win32 are packaged into .intunewin for complex/legacy apps.

45. What tool packages a Win32 app?

The Microsoft Win32 Content Prep Tool (IntuneWinAppUtil.exe), producing a .intunewin file.

46. What are detection rules and why do they matter?

They confirm an app is installed (file/registry/MSI code/script); wrong rules cause incorrect status and repeated installs.

47. How do you deploy Office with Intune?

Use the Microsoft 365 Apps app type (Click-to-Run), choosing apps, 64-bit, and an update channel like Monthly Enterprise.

48. What is an app protection policy?

A MAM policy that secures corporate data in apps (PIN, encryption, copy/paste limits, selective wipe) without managing the device.

49. App protection vs app configuration?

Protection secures app data; configuration pre-sets app settings (server URL, account) so the app works on first launch.

50. How do you update a Win32 app, and how do you troubleshoot a failed install?

Update via supersedence (new version replaces old). Troubleshoot using App install status (error code) and the IntuneManagementExtension.log; suspect detection rules first.

End of the Final Reference Appendix — and of the complete MD-102 study guide.

You now have all four curriculum sections plus this full reference companion. Work the labs, drill the 50 questions, and review the cheat sheets before exam day. Good luck with your MD-102 certification and your endpoint administration career.