JUST IN TIME ANALYSIS

SC-100

CYBERSECURITY ARCHITECT ARCHITECT FIELD GUIDE

Decision Awareness • Risk Context • Microsoft Architecture • Real-World Reasoning

Learn the architecture. Understand the decision. Defend the outcome.

Purpose

This guide teaches the complete reasoning chain behind SC-100: read the situation, identify what matters, assess exposure and business impact, define the control objective, select the correct Microsoft capability, understand dependencies and trade-offs, and validate the architecture end to end.

Created by William S. Davis
Microsoft Certified: Cybersecurity Architect Expert
Just In Time Analysis
Version 3.0 | August 2026

BSDVS23

Publication, Scope & Learning Philosophy

This guide does not reproduce confidential examination questions. All scenarios are original learning scenarios. Microsoft cloud capabilities, licensing, names, and exam objectives change; verify implementation details against current Microsoft documentation.

Important distinction

Frameworks labeled JITA are instructional reasoning models created to help learners think consistently. They are not Microsoft scoring formulas or official Microsoft methodologies unless explicitly identified as Microsoft guidance.

What SC-100 is really testing

Microsoft's current audience profile describes a cybersecurity architect who translates cybersecurity strategy into capabilities protecting organizational assets, business, and operations. The role spans identity, devices, data, AI, applications, networks, infrastructure, DevOps, GRC, security operations, and posture management. That breadth means the exam is fundamentally about architectural judgment across connected systems.

How to use this guide

1. Read each concept until you can explain it without naming a Microsoft product.
2. Work each scenario by identifying the requirement before looking at the technology.
3. Use the decision worksheets to force yourself to state risk, impact, control objective, and enforcement point.
4. Study why an answer is wrong. The distinction between two plausible answers is often the actual lesson.

5. Draw the architecture. If you cannot show trust boundaries, identities, enforcement points, telemetry, dependencies, and recovery paths, your understanding is incomplete.

1 — The Architect's End-to-End Decision Method

The architect's job is not to collect controls. It is to design a coherent system of controls that protects business outcomes while remaining operable, governable, resilient, and proportionate to risk.

Step	Question	Output
1. Business outcome	What must the organization accomplish or preserve?	Outcome statement
2. Facts	Who, what, where, how, and under what conditions?	Evidence set
3. Asset & actor	What is valuable and who/what can act on it?	Asset/identity map
4. Threat/weakness	What can go wrong?	Threat statement
5. Exposure	How reachable or exploitable is the weakness?	Exposure context
6. Impact	What happens to business, data, operations, compliance, or trust?	Impact statement
7. Dominant requirement	Which constraint most changes the design?	Decision driver
8. Principle	Which security principle governs the response?	Zero Trust / least privilege / defense in depth / resilience
9. Control objective	Prevent, detect, respond, recover, govern, or prove?	Control objective
10. Capability	What security function is needed?	IAM/PAM/SIEM/XDR/CNAPP/DLP/etc.
11. Enforcement point	Where must the control operate?	Identity/network/workload/data/code/control plane
12. Microsoft mapping	Which capability best fits?	Primary technology
13. Dependencies	What must also exist?	Supporting controls
14. Trade-offs	What changes in usability, cost, complexity, performance, operations?	Trade-off record
15. Validation	Does the design satisfy every requirement and reduce meaningful risk?	Defensible decision

The most important habit

Do not begin with 'Which Microsoft product is this?' Begin with 'What is the organization trying to accomplish, what can stop it, and what control objective is required?'

2 — How to Read a Scenario Without Missing the Clues

2.1 Build the situation model

Element	Examples	Why it matters
Actor	Employee, admin, guest, workload, service principal, device, attacker	Different actors require different identity and authorization models.
Asset	Tenant, identity, VM, app, database, key, repository, sensitive file	Criticality and control placement depend on the asset.
Action	Sign in, elevate, deploy, share, copy, execute, connect, approve	The verb often reveals the enforcement point.
Condition	Risky, unmanaged, external, public, multicloud, privileged	Conditions narrow the control family.
Constraint	Minimize effort, no public IP, retain access, meet regulation, time-bound	Constraints separate plausible answers.
Evidence	Alert, open port, recommendation, attack path, risky sign-in, data label	Evidence tells you posture vs. active threat vs. governance.
Outcome	Prevent, detect, contain, recover, govern, demonstrate compliance	Outcome identifies the control objective.

2.2 Requirement words that change the answer

Phrase	Architectural interpretation
Just-in-time / eligible / activate / expire	Privilege lifecycle; think PIM.
Compliant device / risk / location / authentication strength	Access policy; think Conditional Access.
Review continued need	Access recertification; think Access Reviews.
Access package / joiner-mover-leaver / entitlement	Identity Governance.
Centralize logs / correlate / hunt / automate SOC	Sentinel / SIEM-SOAR.
Cloud posture / attack paths / multicloud workloads	Defender for Cloud / CNAPP.
Prevent sensitive data sharing	Purview DLP / information protection.
No public RDP/SSH	Bastion.
Open ports only when needed	JIT VM access.
Enforce resource configuration	Azure Policy.
Store secrets / certificates / keys	Key Vault; consider Managed HSM for dedicated HSM requirements.

2.3 Facts vs. assumptions

- A fact is stated or logically required by the scenario.
- A constraint is a fact that eliminates otherwise valid designs.
- An assumption is something common in real environments but not stated.
- Use assumptions to identify implementation questions, not to override the stated requirement.
- If licensing, geography, latency, or legacy compatibility is not stated, do not invent it as the deciding factor unless the question explicitly asks for it.

Common exam failure

Candidates often solve the environment they imagine instead of the environment Microsoft described.

3 — Risk, Threat, Exposure and Business Impact

3.1 Do not use 'severity' as a synonym for risk

A technical weakness becomes strategically important when context makes exploitation plausible and consequences meaningful. Microsoft Defender for Cloud similarly uses environmental context such as internet exposure, data sensitivity, lateral movement, and attack paths to prioritize recommendations.

Dimension	Questions to ask
Asset criticality	Does this asset control identity, security, production, revenue, keys, regulated data, or essential operations?
Exposure	Internet reachable? Broad internal access? Cross-cloud? External sharing?
Privilege	What can the compromised identity/resource change or access?
Exploitability	How many prerequisites exist? Is a direct path available?
Blast radius	One asset, one application, subscription, tenant, enterprise?
Data sensitivity	Public, internal, confidential, regulated, secrets, credentials?
Control strength	Are preventive/detective controls already effective?
Detection time	Would malicious activity be noticed quickly?
Recovery difficulty	Can it be rebuilt, revoked, restored, or rotated quickly?
Business consequence	Outage, fraud, disclosure, legal exposure, safety, reputation, strategic loss?

3.2 JITA contextual priority model

Instructional model
Priority increases when high-value assets, broad exposure, high privilege, exploitable paths, sensitive data, weak controls, and difficult recovery converge. Do not calculate a fake numeric score; use the factors to make the reasoning explicit.

Example	Technical issue	Context	Architectural priority
Dev VM	Open management port	Isolated, no secrets, no production trust path	Important but potentially lower relative priority
Service principal	Long-lived secret	Owner scope, secret in repo, production subscription	Critical candidate
Storage	Overbroad access	Regulated financial data, reachable by many identities	High
Admin account	Weak MFA	Global Administrator, can alter security policy	Critical
Legacy server	Known weakness	Identity infrastructure, reachable from many segments	Critical candidate

3.3 Residual risk matters

Risk after controls are applied is residual risk. A risky sign-in that Conditional Access blocks is not equivalent to a risky sign-in that succeeds. A vulnerability behind strong segmentation is not equivalent to the same vulnerability on an internet-facing critical asset. Architecture decisions should consider both inherent exposure and the effectiveness of existing controls.

4 — Control Objectives: What Must the Architecture Actually Do?

Objective	Meaning	Typical examples
Prevent	Stop an undesired action before it succeeds.	Conditional Access, DLP, policy enforcement, segmentation.
Detect	Identify suspicious behavior, weakness, or deviation.	Defender detections, posture assessments, Sentinel analytics.
Respond	Investigate, contain, automate, and remediate.	XDR investigation, Sentinel automation, incident workflows.
Recover	Restore trusted operation after disruption.	Backups, identity recovery, key recovery, BCDR patterns.
Govern	Keep standards, ownership, exceptions, and accountability consistent.	Azure Policy, identity governance, GRC processes.
Prove	Produce evidence of control operation and compliance.	Audit, compliance reporting, review records, policy state.

4.1 One scenario may need several objectives

A ransomware-resilient architecture may prevent with least privilege and segmentation, detect with XDR/SIEM, respond with automated containment, recover with protected backups and tested recovery, and govern through policy and incident ownership. The primary exam answer depends on the stated requirement, but the real architecture must connect the lifecycle.

Primary vs. complementary
The fact that a second product is useful does not make it the primary answer. Identify the capability that directly satisfies the dominant requirement.

5 — Choosing the Microsoft Capability by Enforcement Point

Control point	Primary questions	Microsoft direction
Identity sign-in	Should access be allowed under these conditions?	Entra Conditional Access / Identity Protection / authentication controls
Privileged activation	Should this identity hold elevated authority right now?	Entra PIM
Access lifecycle	Should this person receive/retain this entitlement?	Entra ID Governance / Access Reviews
Workload identity	How should software authenticate and be authorized?	Managed identities / workload identities
Endpoint / identity / email / SaaS detection	Is a coordinated attack occurring?	Microsoft Defender XDR ecosystem
SOC	How do we aggregate, correlate, hunt, and orchestrate?	Microsoft Sentinel
Cloud posture/workloads	Where are cloud risks and exploitable paths?	Defender for Cloud / CNAPP
Azure resource configuration	Does the resource meet organizational standards?	Azure Policy
Network / edge	What traffic should be allowed, inspected, or protected?	NSG / Firewall / WAF / DDoS depending on layer
Administration path	How should administrators reach VMs?	Bastion / JIT depending on requirement
Data	How is sensitive data classified, protected, governed, and prevented from leaking?	Microsoft Purview capabilities
Code / pipeline	How do we find weaknesses before deployment?	GitHub security / DevSecOps controls
Secrets / keys	How are credentials and cryptographic material protected?	Key Vault / Managed HSM

5.1 The 'wrong layer' test

A common distractor solves the right general security theme at the wrong layer. Sentinel can alert on a configuration problem but does not replace Azure Policy enforcement. Conditional Access can restrict access to an app but does not replace DLP for controlling sensitive content movement. Defender for Cloud can identify posture risk but does not replace PIM for privileged role activation.

6 — Identity Architecture End to End

6.1 Human identity lifecycle

6. Establish identity and authoritative source.
7. Provision only required access.
8. Apply authentication strength and access policy based on risk and context.
9. Separate standard and privileged activity.
10. Make privilege eligible/time-bound where appropriate.
11. Review access periodically and on lifecycle events.
12. Detect identity compromise and suspicious behavior.
13. Remove access promptly when business need ends.

6.2 Conditional Access

Conditional Access is Microsoft's Zero Trust policy engine. It evaluates signals and applies decisions. Think in IF-THEN logic: if this identity accesses this resource under these conditions, then require, restrict, or block.

Signal / condition	Possible architectural use
User/group	Target population or administrative role.
Cloud app/resource	Protect a specific application or resource.
Device state/compliance	Require managed/compliant device conditions.
Sign-in/user risk	Adapt policy to identity risk.
Location/network	Apply network/location conditions where appropriate.
Authentication strength	Require stronger methods for sensitive access.

6.3 PIM

PIM mitigates excessive, unnecessary, or misused privileged access through eligible and time-bound assignments, activation controls, approval, MFA/Conditional Access integration, justification, notifications, and auditing.

Remember
MFA does not remove standing privilege. PIM changes the privilege lifecycle; Conditional Access changes the access conditions.

6.4 Identity Governance and Access Reviews

Identity Governance answers who should have access, to what, for how long, through what approval/lifecycle process, and whether that access should continue. Access Reviews are a recertification mechanism within that broader governance problem.

6.5 Workload identities

Applications, automation, service principals, managed identities, and AI agents can hold powerful permissions. Treat non-human identities as first-class security subjects: minimize scope, avoid long-lived secrets, monitor use, govern ownership, and rotate/revoke credentials where applicable.

Identity scenario

A deployment pipeline needs to create resources in one resource group. It currently uses a client secret for a service principal with Contributor on the subscription.

14. Asset: production subscription and deployment pipeline.
15. Weakness: long-lived secret plus excessive scope.
16. Impact: secret compromise could affect resources outside the pipeline's need.
17. Principle: least privilege and credential minimization.
18. Design: use workload/managed identity where supported and scope authorization to the required resource group/operations.
19. Validation: confirm the pipeline still performs required deployment actions without broader authority.

7 — Security Operations, XDR, SIEM and Exposure Management

7.1 Know the operating question

Question	Primary capability
Is there a coordinated attack across identities/endpoints/email/apps?	Defender XDR
What is happening across Microsoft and third-party telemetry?	Sentinel
Where are cloud posture weaknesses and attack paths?	Defender for Cloud
Which security weaknesses create the most meaningful exposure?	Risk/exposure prioritization capabilities
What should the SOC automate after a detection?	Sentinel/XDR response workflows depending on scenario

7.2 Posture is not the same as incident response

Posture evidence	Incident evidence
Open management port	Observed brute-force/exploitation activity
Overprivileged identity	Suspicious role activation or token abuse
Missing endpoint protection	Malware execution
Public storage configuration	Observed unauthorized access/exfiltration
Unpatched vulnerability	Exploit behavior or compromise indicators

7.3 Context-aware prioritization

Microsoft Defender for Cloud's risk prioritization uses environmental context including resource configuration, network connections, internet exposure, data sensitivity, lateral movement, and attack paths. The lesson for an architect is broader: prioritize the weakness that creates the most credible path to the most important consequence.

7.4 Telemetry architecture

- Decide what must be logged before selecting retention and analytics.
- Protect log integrity and privileged access to security tooling.
- Normalize ownership: who triages, who investigates, who remediates?
- Define escalation for identity, cloud, endpoint, data, and application incidents.
- Design automation for high-confidence actions; avoid automation that can create uncontrolled business impact.
- Measure mean time to detect, investigate, contain, and recover where operationally meaningful.

8 — Infrastructure, Network and Management Plane Security

8.1 Management plane first

If an attacker can alter roles, policies, keys, logging, networking, or security controls, downstream protections can be weakened. Protect the management plane with strong identity, PIM, Conditional Access, separation of duties, logging, and tightly scoped authorization.

8.2 Network decision ladder

Question	Control direction
Should this subnet/NIC traffic be allowed?	NSG / network segmentation.
Do we need centralized managed firewall policy/inspection?	Azure Firewall.
Do we need HTTP/S application-layer protection?	WAF.
Do we need DDoS resilience for public resources?	Azure DDoS Protection.
Do admins need RDP/SSH without public management endpoints?	Azure Bastion.
Should management ports open only temporarily?	JIT VM access.
Can private connectivity remove public exposure?	Private networking/private endpoints where supported.

8.3 Keys and secrets

Ask whether the scenario involves secrets, certificates, encryption keys, dedicated HSM requirements, key ownership, rotation, recovery, or separation of duties. Key Vault is the broad platform for secrets/certificates/keys; Managed HSM addresses dedicated single-tenant HSM-backed key-control requirements.

8.4 Recovery architecture

- Identify recovery dependencies: identity, DNS/network, keys, data, applications, monitoring.
- Protect backups from the same identities and attack paths that can destroy production.
- Define recovery order, RTO/RPO, and validation of restored trust.
- Test restoration; a backup that has never been restored is an assumption, not a recovery capability.
- Plan for compromised credentials and keys, not only failed servers.

9 — Data Security and Microsoft Purview End to End

9.1 Follow the data lifecycle

Stage	Questions
Discover	What data exists and where?
Classify	What is sensitive, regulated, confidential, or public?
Protect	What labels, encryption, rights, or access controls apply?
Use	Who/what may process it and under which conditions?
Share	Where may it move? What should be blocked/warned/audited?
Retain	How long must it exist?
Investigate	Can activity be audited, searched, and preserved?
Dispose	How is deletion governed and verified?

9.2 Purview capability reasoning

Requirement	Think first about
Recognize and label sensitive content	Information Protection / sensitivity labels.
Prevent inappropriate data movement	DLP.
Restrict collaboration between defined groups	Information Barriers.
Retain/dispose according to policy	Data Lifecycle / Records Management.
Search/preserve content for legal matters	eDiscovery.
Investigate risky internal behavior	Insider Risk Management where appropriate.
Audit activity	Audit capabilities.

Critical distinction

Authentication answers 'may this identity access the resource?' Data protection answers 'what may this authorized identity do with this specific information?' Mature architecture often needs both.

10 — Application, API, DevSecOps and AI Security

10.1 Secure the software lifecycle

Stage	Controls to reason about
Design	Threat modeling, trust boundaries, identity model, data flows, least privilege.
Code	Secret scanning, code analysis, dependency awareness, secure patterns.
Build	Protected pipelines, trusted artifacts, least-privileged build identities.
Deploy	Policy, IaC validation, controlled configuration, workload identity.
Runtime	WAF/API controls, workload protection, telemetry, secrets, segmentation.
Operate	Patch, detect, investigate, rotate, recover, govern.

10.2 API security

- Identify caller identity and authentication method.
- Authorize at the narrowest appropriate scope.
- Validate input and enforce schema/business rules.
- Protect secrets and tokens.
- Rate-limit and monitor abuse where appropriate.
- Log security-relevant activity without exposing sensitive data.
- Consider WAF/API management/runtime controls based on the architecture.

10.3 AI security reasoning

The 2026 SC-100 audience profile explicitly includes AI security. Treat AI systems as connected applications with identities, data, tools, permissions, and trust boundaries.

- What identity does the model, agent, connector, or tool use?
- What data can it retrieve, infer, transform, or disclose?
- Can the agent take actions, or only recommend them?
- What approval boundary exists before privileged or irreversible actions?
- How are prompts, outputs, tool calls, and data access logged?
- Can untrusted content influence tool execution or data retrieval?
- How are data classification, DLP, access control, and governance extended to AI workloads?

11 — Governance, GRC and Microsoft Architecture Frameworks

11.1 Governance is how architecture survives operations

Governance question	Required design element
Who owns the control?	Accountable role/team.
Where is it enforced?	Technical/process enforcement point.
How is effectiveness measured?	Telemetry, assessment, audit, KPIs/KRIs.
Who may approve exceptions?	Defined authority and risk acceptance.
When does an exception expire?	Time-bound exception lifecycle.
What compensating control is required?	Alternative risk reduction.
How is drift remediated?	Policy, automation, operational workflow.
How does leadership see risk?	Business-aligned reporting and prioritization.

11.2 MCRA and security adoption

Microsoft's MCRA provides end-to-end technical architectures for Zero Trust modernization across a 'hybrid of everything' estate including legacy IT, multicloud, OT/IoT, and AI. Microsoft positions MCRA as part of a structured security adoption model that helps organizations plan, prioritize, design, and implement modernization.

11.3 Do not confuse the frameworks

Framework/capability	Best mental model
MCRA	What does the end-to-end security architecture look like and how do capabilities integrate?
Zero Trust	What principles govern trust and access?
Security adoption model	How do we plan/prioritize/implement modernization across the organization?
Cloud Adoption Framework	How do we organize broader cloud adoption, governance, management, and security?
Well-Architected Framework	How do we make workload-level design trade-offs across pillars?
Microsoft Cloud Security Benchmark	What cloud security controls/recommendations should guide posture?
Azure Policy	How do we evaluate/enforce resource configuration standards at scale?

12 — Things That Get Overlooked

Overlooked issue	Why it matters
Standing privilege	MFA protects authentication but does not eliminate persistent authority.
Service principals / managed identities	Non-human identities can have enormous blast radius.
Scope	A correct role at the wrong scope is still excessive privilege.
Control dependencies	A policy may depend on device compliance, identity signals, logging, or licensing.
Break-glass accounts	Emergency access must be protected, tested, excluded carefully, and monitored.
Security tooling privilege	Attackers who control security policy/logging can blind defenders.
Attack paths	Individual findings become more important when chained together.
Data sensitivity	A moderate weakness protecting regulated data may outrank a severe issue on a low-value asset.
Recovery identities/keys	Recovery fails if the same compromised identities control backups and keys.
Operational ownership	A control nobody owns becomes shelfware or permanent exception.
Exceptions	Permanent exceptions quietly become architecture.
User experience	Controls that create unusable workflows invite bypass and shadow IT.
Multicloud scope	Azure-only governance may not satisfy enterprise multicloud requirements.
Detection without remediation	Visibility is not risk reduction until action occurs.
Compliance without threat context	A passed control does not prove an attacker lacks a viable path.

13 — Detailed Decision Labs

Lab 1: Privileged identity

Situation: Admins are permanent Global Administrators. Require approval, phishing-resistant authentication, justification, and 60-minute elevation.

Decision
Primary: PIM for eligible/time-bound privilege + Conditional Access/authentication strength for access conditions.

Why: privilege lifecycle and authentication conditions are two different control objectives. Access Reviews may recertify assignments but do not replace per-use activation.

Lab 2: Multicloud posture

Situation: Azure and AWS workloads need one prioritized view of posture weaknesses, attack paths, and workload protection.

Decision
Primary: Defender for Cloud / CNAPP.

Why: the dominant requirement is multicloud posture and workload context. Sentinel is complementary when SOC correlation is required.

Lab 3: SOC centralization

Situation: Security needs telemetry from Microsoft, Linux, network appliances, and third-party cloud systems with hunting and automated response.

Decision
Primary: Microsoft Sentinel.

Why: heterogeneous telemetry + SIEM/SOAR functions define the requirement.

Lab 4: Sensitive data

Situation: Remote users may access financial documents, but external sharing and copying to unmanaged destinations must be controlled.

Decision
Primary: Purview information protection/DLP + access/device controls as needed.

Why: Conditional Access governs entry; DLP governs sensitive data movement after entry.

Lab 5: VM administration

Situation: VMs must have no public RDP/SSH exposure but administrators require interactive access.

Decision
Primary: Azure Bastion.

Why: no-public-management-path is the dominant requirement. JIT is about duration of port exposure.

Lab 6: Resource governance

Situation: Every new storage account must meet approved configuration standards with centralized compliance reporting.

Decision
Primary: Azure Policy.

Why: the requirement is resource configuration governance/enforcement, not merely detection.

Lab 7: Credential in repository

Situation: A CI/CD pipeline stores a secret for a service principal with subscription-wide Contributor.

Decision
Primary: redesign toward workload identity/managed identity where supported, narrow RBAC scope, repository secret scanning, and Key Vault only where secrets remain necessary.

Why: simply moving an overprivileged long-lived secret into a vault does not solve excessive authorization or credential dependence.

Lab 8: Attack-path priority

Situation: Ten recommendations exist. One moderate recommendation is on an internet-exposed resource that can laterally reach a critical identity system.

Decision
Primary: prioritize the contextual attack path, not the label alone.

Why: Microsoft Defender for Cloud risk prioritization considers exposure, lateral movement, data sensitivity, attack paths, and potential business impact.

14 — Complete Architecture Decision Framework

The objective is not to memorize this framework. It is to develop the habit of asking these questions automatically whenever you evaluate a security architecture. Each decision area below represents a point of awareness that should be resolved before the final architecture is selected.

Decision Area	What You Need to Determine
Business Outcome	What business capability, operation, or objective must be protected or enabled?
Actors / Identities	Which users, administrators, workloads, service principals, devices, guests, or external identities are involved?
Critical Assets	Which identities, systems, applications, data, keys, security tools, or services would create the greatest impact if compromised?
Trust Boundaries	Where does trust change between identities, devices, networks, applications, clouds, administrative planes, and data?
Threat / Weakness	What could realistically be exploited, abused, misconfigured, stolen, or compromised?
Evidence	What facts, alerts, recommendations, configurations, or behaviors establish that the issue exists?
Exposure / Reachability	How reachable is the weakness, from where, and through which path?
Privilege / Authorization	What authority exists now, and what authority could be obtained or abused?
Data Sensitivity	What information could be exposed, altered, destroyed, or misused, and how sensitive is it?
Blast Radius	If the control fails or compromise succeeds, how far can the impact spread?
Existing Controls	What currently prevents, detects, limits, or responds to the threat?
Residual Risk	After existing controls operate as intended, what meaningful risk remains?
Dominant Requirement	Which business, security, compliance, operational, or architectural requirement most constrains the decision?
Security Principle	Which principle should govern the design: Zero Trust, least privilege, assume breach, defense in depth, secure by design, segmentation, resilience, or governance?
Control Objective	Does the architecture primarily need to prevent, detect, respond, recover, govern, or prove?
Capability Category	What type of security capability is required before selecting a Microsoft product?
Enforcement Point	Where must the control operate: identity, privilege, endpoint, network, workload, application, data, code/pipeline, or management plane?
Primary Microsoft Capability	Which Microsoft capability most directly satisfies the dominant requirement?
Complementary Capabilities	Which additional controls strengthen the design without replacing the primary capability?

Dependencies / Prerequisites	What identities, telemetry, policies, agents, device state, network design, licensing, integrations, or operational processes must exist?
Operational Impact	What changes for users, administrators, developers, the SOC, application teams, or business operations?
Governance / Ownership	Who owns the control, monitors it, approves exceptions, and is accountable for remediation?
Exception Management	What qualifies as an exception, who approves it, what compensating control is required, and when does the exception expire?
Failure Mode	What happens if the control is unavailable, bypassed, misconfigured, or compromised?
Recovery Requirement	How does the organization restore trusted operations, identities, data, keys, and services?
Trade-offs	What security, usability, performance, cost, complexity, resilience, or operational trade-offs are introduced?
Effectiveness Measure	How will the organization know the control is operating correctly and risk is actually decreasing?
Final Architecture Decision	What design provides the strongest appropriate risk reduction while satisfying the stated business requirements and avoiding unnecessary complexity?

15 — Capstone: Build the Architecture, Then Prioritize It

Fabrikam operates Microsoft 365, on-premises AD, Azure, AWS, GitHub, and several AI-enabled internal applications. Findings include permanent administrators, a service principal secret in source code, inconsistent Conditional Access, sensitive financial data, fragmented SOC telemetry, public VM management ports, and no standardized multicloud posture process.

Phase 1 — Prioritize

20. Identify crown-jewel assets: identity control plane, financial data, production applications, keys, security tooling.
21. Map privileged identities and non-human identities.
22. Identify internet exposure and lateral movement paths.
23. Identify which findings can alter security controls or reach high-value assets.
24. Account for existing controls and residual risk.
25. Rank remediation by contextual business risk, not by product or severity label alone.

Phase 2 — Design

Need	Architecture direction
Privileged administration	PIM + Conditional Access + scoped RBAC + monitoring.
Adaptive user access	Conditional Access + strong authentication + device/risk signals.
Access lifecycle	Identity Governance / Access Reviews.
Multicloud posture/workloads	Defender for Cloud.
Central SOC	Sentinel integrated with Defender XDR and relevant sources.
Sensitive data	Purview classification/protection/DLP/governance.
Repository and pipeline security	GitHub security + workload identity + least-privileged RBAC.
VM management	Bastion/JIT based on exact exposure requirement.
Resource standards	Azure Policy and governance hierarchy.
Target architecture	MCRA/Zero Trust/security adoption guidance as reference patterns.

Phase 3 — Operate and govern

- Assign control owners and remediation owners.
- Define telemetry and escalation paths.
- Track exceptions with expiry and compensating controls.
- Measure risk reduction, not only deployment completion.
- Test privileged workflows, incident response, and recovery.
- Review architecture as identities, data, clouds, AI agents, and business requirements change.

Capstone lesson

The architecture is not the list of products. The architecture is the relationship among identities, permissions, assets, trust boundaries, controls, telemetry, operations, governance, and recovery.

16 — Exam Alignment and Preparation

Skills measured as of July 28, 2026	Weight
Design solutions that align with security best practices and priorities	20–25%
Design security operations, identity, and compliance capabilities	25–30%
Design security solutions for infrastructure	25–30%
Design security solutions for applications and data	20–25%

Under exam pressure

- Read the requirement before the answer choices.
- Identify the verb: prevent, require, detect, review, govern, recover, minimize.
- Identify the noun: identity, privilege, data, cloud posture, telemetry, network, key, workload.
- Find the constraint that eliminates otherwise correct answers.
- Separate primary capability from complementary capability.
- Check the enforcement point.
- Prefer the answer satisfying all explicit requirements with the least unnecessary complexity.
- Do not confuse a monitoring capability with an enforcement capability.
- Do not confuse a posture recommendation with an active incident.
- Do not confuse compliance evidence with complete security.

17 — Final Mastery Checklist

- ☐ I can translate a business outcome into a security requirement.
- ☐ I can distinguish threat, weakness, exposure, impact, risk, incident, and residual risk.
- ☐ I can explain why asset criticality, privilege, internet exposure, lateral movement, data sensitivity, and attack paths change priority.
- ☐ I can distinguish Conditional Access, PIM, Identity Governance, Access Reviews, and workload identities.
- ☐ I can distinguish Defender XDR, Sentinel, Defender for Cloud, Azure Policy, and exposure/posture concepts.
- ☐ I can distinguish Bastion from JIT, NSG from Firewall, and WAF from DDoS protection.
- ☐ I can follow data from discovery through classification, protection, DLP, retention, investigation, and disposal.
- ☐ I can explain build-time versus runtime application security.
- ☐ I can include AI identities, data, tools, and permissions in a security architecture.
- ☐ I can explain MCRA, Zero Trust, security adoption guidance, CAF, WAF, MCSB, and Azure Policy without treating them as synonyms.
- ☐ I can identify primary versus complementary controls.
- ☐ I can explain the dependencies and failure modes of my design.
- ☐ I can define how the control is owned, measured, excepted, and recovered.
- ☐ I can defend why my architecture is appropriate for the stated business risk.

Official Microsoft References

- **SC-100 Study Guide:** https://learn.microsoft.com/en-us/credentials/certifications/resources/study-guides/sc-100
- **Exam SC-100:** https://learn.microsoft.com/en-us/credentials/certifications/exams/sc-100/
- **Microsoft Cybersecurity Reference Architectures:** https://learn.microsoft.com/en-us/security/zero-trust/microsoft-reference-architecture
- **Microsoft security adoption model:** https://learn.microsoft.com/en-us/security/zero-trust/security-adoption-model
- **Zero Trust adoption framework overview:** https://learn.microsoft.com/en-us/security/zero-trust/adopt/zero-trust-adoption-overview
- **Conditional Access overview:** https://learn.microsoft.com/en-us/azure/active-directory/conditional-access/overview
- **Plan a PIM deployment:** https://learn.microsoft.com/en-us/entra/id-governance/privileged-identity-management/pim-deployment-plan
- **Microsoft Entra ID Governance:** https://learn.microsoft.com/en-us/entra/id-governance/identity-governance-overview
- **Access Reviews overview:** https://learn.microsoft.com/en-us/entra/id-governance/access-reviews-overview
- **Defender for Cloud security recommendations:** https://learn.microsoft.com/en-us/azure/defender-for-cloud/security-recommendations
- **Defender for Cloud risk prioritization:** https://learn.microsoft.com/en-us/azure/defender-for-cloud/risk-prioritization
- **Defender for Cloud attack paths:** https://learn.microsoft.com/en-us/azure/defender-for-cloud/how-to-manage-attack-path
- **Defender for Cloud secure score:** https://learn.microsoft.com/en-us/azure/defender-for-cloud/secure-score-security-controls

Closing principle

See the system, not just the product. The strongest architect can explain what matters, why it matters, where the control belongs, how the capabilities connect, what can fail, how the organization recovers, and how the decision reduces business risk.

William S. Davis
Just In Time Analysis

BSDVS23

18 — Field Guide: Microsoft Decision Patterns in Depth

This section is intentionally written the way I would walk someone through the architecture. I want you to understand what changes the decision, not memorize a service name.

18.1 Private Endpoint vs. NSG vs. Service Endpoint

What You Need to Understand

These controls all affect network access, but they solve different problems. An NSG filters IP/port/protocol traffic at subnet or NIC boundaries. A private endpoint gives a supported PaaS resource a private IP in your virtual network through Azure Private Link. Service endpoints extend virtual-network identity to supported Azure services while the service still uses its public endpoint.

How I Look at It

Before I choose anything, I ask what Microsoft is trying to remove. Is the problem unwanted network flows inside the virtual network? Is it public exposure of a PaaS service? Or is the requirement simply to restrict a service to selected virtual networks? Those are not the same problem.

What I Want You to Notice

Look for wording such as 'private IP,' 'disable public network access,' 'access the PaaS resource privately,' versus 'allow only TCP 443 between these subnets.' The first set points toward Private Link/private endpoints; the latter is network filtering.

Decision Logic

Map the enforcement point. NSG = network flow filtering. Private Endpoint = private connectivity to a specific service instance. Service Endpoint = service access restricted to selected VNets/subnets without placing the service itself on a private IP in your VNet.

What People Miss
People often choose NSG because the scenario says 'network security.' An NSG does not magically make a public PaaS endpoint private.

Change One Fact, Change the Decision
If the scenario changes from 'storage must have no public network exposure' to 'only AppSubnet may connect to DbSubnet on TCP 1433,' the decision changes from private connectivity to traffic filtering.

The Bigger Picture

Real architectures frequently combine controls: private endpoint for the service path, DNS configured correctly, NSGs/segmentation around workloads, identity authorization at the service, and logging to verify use.

18.2 Azure Bastion vs. Just-in-Time VM Access

What You Need to Understand

Bastion and JIT both reduce management exposure, but the architectural intent differs. Bastion provides a managed RDP/SSH access path without requiring public IP-based management exposure. JIT limits the duration during which selected management ports can be opened.

How I Look at It
I do not see 'RDP' and immediately choose Bastion. I read what Microsoft wants changed. If the requirement says no public RDP/SSH endpoint but interactive administration is still required, Bastion fits. If the requirement says ports should be closed by default and opened only for an approved maintenance window, JIT fits.

What I Want You to Notice
The clue is not the protocol. The clue is the exposure model.

Decision Logic

Choose the control that changes the exact condition Microsoft described. Then determine whether the other control is complementary.

What People Miss
A candidate can know both products perfectly and still miss the question by solving the wrong exposure problem.

Change One Fact, Change the Decision
Change 'never expose public management ports' to 'permit RDP for 30 minutes after approval' and the primary answer changes.

The Bigger Picture

This is a recurring SC-100 pattern: two technologies can both improve security, but only one directly satisfies the dominant requirement.

18.3 Key Vault vs. Managed HSM

What You Need to Understand

Key Vault broadly manages secrets, certificates, and cryptographic keys. Managed HSM is focused on highly controlled, dedicated single-tenant HSM-backed cryptographic keys. The decision is driven by the cryptographic custody and isolation requirement, not by the mere appearance of the word 'key.'

How I Look at It

I first separate a secret from a key. Then I ask whether the requirement is application secret/certificate/key management or a dedicated HSM-backed key-control boundary with stronger isolation expectations.

What I Want You to Notice

Watch for compliance or key-custody language that specifically changes the cryptographic boundary.

Decision Logic

Use Key Vault for broad secrets/certificates/keys scenarios. Consider Managed HSM when the requirement specifically calls for dedicated HSM-backed key management and its operational model.

What People Miss

Moving a secret into a vault does not fix an application that still has excessive authorization. Storage of the credential and scope of the identity are separate decisions.

Change One Fact, Change the Decision

If the requirement changes from 'store database credentials securely' to 'organization requires dedicated single-tenant HSM-backed control of cryptographic keys,' the architectural direction changes.

The Bigger Picture

Cryptography is part of a larger chain: identity controls who can manage keys, networking controls how the service is reached, logging records use, recovery protects availability, and governance defines custody and separation of duties.

18.4 Always Encrypted vs. TDE and Database Protection

What You Need to Understand

Encryption at rest and protection from privileged database access are different threat models. Transparent Data Encryption protects database files, backups, and logs at rest. Always Encrypted is designed so sensitive values can remain encrypted from the database engine, with encryption/decryption occurring through the client driver under the appropriate key architecture.

How I Look at It

When I see an encryption question, I ask: 'Protected from what?' Disk theft? Backup exposure? Database administrators? Application operators? The threat model decides the control.

What I Want You to Notice

If Microsoft says sensitive columns must remain unreadable to database administrators, simple at-rest encryption does not satisfy that trust boundary.

Decision Logic

TDE addresses storage-at-rest protection. Always Encrypted addresses sensitive column protection where the database engine should not see plaintext, subject to supported query/functionality considerations and key design.

What People Miss

'Encrypt the database' is not enough information. You need to know who is inside and outside the trust boundary.

Change One Fact, Change the Decision

If the requirement changes from 'protect database files and backups at rest' to 'DBAs must not see cardholder values,' the decision changes.

The Bigger Picture

This is the larger architecture lesson: encryption technology is selected according to who you do and do not trust, where keys live, where plaintext exists, and what operations must remain possible.

18.5 Azure Backup, Resource Guard, Immutability and Ransomware

What You Need to Understand

Backup security is not just about whether a backup exists. You must consider whether an attacker who compromises production administrators can also delete, disable, or weaken the recovery system. Resource Guard and multi-user authorization patterns add protection around critical backup operations; immutability and soft-delete-related capabilities strengthen recovery against malicious or accidental destruction.

How I Look at It

I treat backup as a separate security boundary. If the same compromised administrator can destroy production and erase recovery, I do not really have ransomware resilience.

What I Want You to Notice

Pay attention to destructive operations: disabling protection, deleting recovery points, weakening immutability, changing vault controls, or modifying security settings.

Decision Logic

Design backup with separation of duties, protected critical operations, immutability where required, strong identity, monitoring, and tested recovery. Choose the specific Azure Backup capability based on the stated destructive action or recovery requirement.

What People Miss

People focus on RPO/RTO and forget adversarial recovery. Cyber recovery assumes someone may intentionally attack the recovery mechanism.

Change One Fact, Change the Decision

If the scenario changes from accidental deletion to 'a compromised subscription administrator must be prevented from deleting backups without a second security boundary,' Resource Guard/multi-user authorization becomes much more important.

The Bigger Picture

Recovery is part of Zero Trust's assume-breach posture. Identity recovery, key recovery, data recovery, and application recovery must be designed together and tested.

18.6 Azure Policy vs. Defender for Cloud

What You Need to Understand

Azure Policy evaluates and can enforce resource configuration standards. Defender for Cloud provides cloud security posture management, recommendations, contextual prioritization, attack paths, and workload protection capabilities. They integrate, but they are not interchangeable.

How I Look at It

I ask whether the organization wants to define/enforce the standard or understand/prioritize cloud security risk. 'Every storage account must deny public access' sounds like policy enforcement. 'Which weaknesses create the most dangerous attack path across Azure and AWS?' sounds like posture/exposure management.

What I Want You to Notice

One is primarily governance/enforcement of resource state; the other adds security posture and workload-risk context.

Decision Logic

Use Azure Policy for standardized resource configuration governance. Use Defender for Cloud when the requirement is CSPM/CNAPP, security recommendations, contextual risk, attack paths, or workload protection.

What People Miss

A Defender recommendation can tell you something is wrong. That does not mean Defender is the mechanism enforcing the organization's desired configuration.

Change One Fact, Change the Decision

Change 'deny creation of noncompliant resources' to 'prioritize multicloud weaknesses by attack path and business risk' and the answer changes.

The Bigger Picture

Strong cloud governance often uses both: policy establishes guardrails, Defender for Cloud supplies security context and prioritization, and operations closes the remediation loop.

18.7 Entra Cross-Tenant Access

What You Need to Understand

Cross-tenant access settings govern inbound and outbound B2B collaboration/direct-connect trust relationships between Microsoft Entra organizations. The architect must distinguish external identity lifecycle, authentication trust, tenant boundaries, and resource authorization.

How I Look at It

I do not treat 'external user' as one generic condition. I want to know which tenant authenticates the user, which tenant owns the resource, what trust is accepted, whether MFA/device claims are trusted, and how access is governed over time.

What I Want You to Notice

Tenant boundary, direction of trust, and lifecycle are the clues. External collaboration is both an identity architecture and governance problem.

Decision Logic

Use cross-tenant access settings to define organizational trust behavior; combine with Conditional Access, entitlement management/access reviews, and least-privileged resource authorization as required.

What People Miss

Trusting an MFA claim from another tenant is not the same as granting access to your resources. Authentication trust and authorization are separate.

Change One Fact, Change the Decision

If the partner becomes an acquired subsidiary with different governance requirements, the trust, lifecycle, and tenant architecture may need to change even if the users are the same people.

The Bigger Picture

External identity architecture should answer who authenticates, who authorizes, who reviews, who revokes, what claims are trusted, and what happens when the relationship ends.

18.8 Enterprise-Scale Landing Zones and Security Governance

What You Need to Understand

A landing zone is not merely a subscription template. It establishes scalable foundations for identity, management groups, subscriptions, policy, networking, management, security, and platform governance.

How I Look at It

When I see rapid cloud growth, inconsistent subscriptions, or business units building their own patterns, I think beyond individual controls. I ask whether the operating model itself needs a governed platform foundation.

What I Want You to Notice
Management-group hierarchy, policy inheritance, connectivity, centralized security, ownership, and subscription vending are architecture signals.

Decision Logic

Use landing-zone/CAF patterns to establish scalable platform governance, then layer workload-specific Well-Architected decisions within that governed environment.

What People Miss
Deploying Defender or Policy after every team builds differently is not the same as designing a governed cloud platform.

Change One Fact, Change the Decision
If the organization has one isolated workload, enterprise-scale platform patterns may be unnecessary overhead. Change the scale and organizational complexity, and the appropriate architecture changes.

The Bigger Picture

This is where cybersecurity architecture meets enterprise architecture: controls must scale through hierarchy, automation, ownership, policy, and repeatable platform patterns.

18.9 GitHub, Branch Protection and DevSecOps

What You Need to Understand

DevSecOps security spans repository governance, branch protections, code and secret scanning, dependencies, build identities, environments, approvals, artifacts, and runtime deployment.

How I Look at It
I want to know where the risk enters the software supply chain. Is someone bypassing review? Is a secret committed? Is a dependency vulnerable? Can the pipeline deploy directly to production? Does the pipeline identity have too much authority?

What I Want You to Notice
Repository, pipeline, environment, and runtime are different trust boundaries.

Decision Logic

Choose controls at the stage where the risk occurs: branch protection/review for unauthorized code change, secret scanning for committed credentials, code/dependency scanning for weaknesses, protected environments/approvals for deployment, workload identity and narrow RBAC for pipeline access.

What People Miss
Scanning code does not solve an overprivileged deployment identity. Requiring pull requests does not remove secrets already embedded in automation.

Change One Fact, Change the Decision
If the issue changes from 'developers bypass review' to 'pipeline secret has Owner rights,' the control moves from branch governance to identity/authorization design.

The Bigger Picture

The secure software supply chain is a chain. Attackers only need one weak link; architects must understand how repository identity, build identity, artifacts, deployment authorization, runtime, and monitoring connect.

18.10 Entra Internet Access and Modern Access Architecture

What You Need to Understand

Microsoft Entra Internet Access is part of Microsoft's Security Service Edge/Global Secure Access direction and applies identity-centric access controls to internet/SaaS traffic. The architectural question is whether access control should follow identity and context rather than depend solely on traditional network location.

How I Look at It

I look for a shift from 'inside network equals trusted' to 'every access request is evaluated using identity, device, destination, and policy.' That is a Zero Trust access architecture decision.

What I Want You to Notice

Do not confuse internet access security with private application access. Identify whether the destination is public internet/SaaS or private organizational resources.

Decision Logic

Map the requirement to the appropriate Global Secure Access capability and integrate with Conditional Access and identity signals where supported.

What People Miss

Network security and identity security are converging, but they are not identical. You still need to know where traffic goes and where policy is enforced.

Change One Fact, Change the Decision

If the destination changes from public SaaS to a private legacy application, the access architecture changes.

The Bigger Picture

This is an example of why SC-100 is architectural: identity, network, device posture, SaaS use, and policy increasingly operate as one access system.

19 — My Decision Review Before I Commit to an Answer

Before I commit to an architecture decision, I mentally challenge it. I want you to develop the same habit.

Challenge	What I ask myself
Requirement	Did I solve what was actually asked, or what I assumed was asked?
Risk	What happens if I do nothing? What is the realistic business consequence?
Control layer	Am I enforcing at the correct point, or using a product from the right category at the wrong layer?
Scope	Is the control applied narrowly enough to preserve least privilege but broadly enough to satisfy the requirement?
Primary vs. complementary	Which capability directly solves the problem, and which ones merely strengthen it?
Dependency	What must be true for this control to work? Identity signal? DNS? Agent? Device compliance? Logging?
Attacker view	Can an attacker bypass this by compromising the administrator, identity, key, pipeline, or recovery system?
Operations	Who owns this after deployment? Who responds when it fails?
Recovery	Can I restore trusted operation, not merely restore data?
Trade-off	What did I make harder, more expensive, slower, or more complex? Is that acceptable?
Change test	What single fact would cause me to choose a different architecture?

My standard for you

I do not want you leaving this guide knowing only that PIM means privileged access or Sentinel means SIEM. I want you able to walk into an unfamiliar situation, identify what matters, understand the degree of risk, choose the correct control layer, explain the Microsoft decision, and defend why the architecture should change when the facts change.