A Unified Enterprise Video Curriculum

Rule-BAC • RBAC • ABAC • MAC • DAC

Evidence-led instructional design blueprint • detailed illustrated edition

> The core idea: treat the five models as **policy lenses that can operate simultaneously on one access request**.

This edition preserves the approved research, linked citations, scenarios, assessment method, measurement plan, limitations, and complete source ledger.

Executive answer

An effective enterprise curriculum should not teach Rule-Based Access Control (Rule-BAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), Mandatory Access Control (MAC), and Discretionary Access Control (DAC) as five competing products or as a vocabulary list. It should teach them as **policy lenses that can operate simultaneously on one access request**:

- **DAC** asks what the resource owner is allowed to share or delegate.
- **RBAC** asks what permissions follow from an approved organizational function.
- **ABAC** evaluates subject, object, action, and environmental attributes against policy.
- **MAC** imposes centrally controlled labels and rules that the user or owner cannot override.
- **Rule-BAC** applies organization-defined conditional rules; in practice it may complement RBAC, resemble ABAC, or implement mandatory policy.

This framing matters because “Rule-BAC” does not have the same stable formal meaning as the other labels. NISTIR 7316 explicitly says there was no commonly understood definition or formal standard for Rule-Based Access Control comparable to DAC, MAC, and RBAC, and describes it as a generic term for systems using organization-defined rules. It also says rules can replace or complement RBAC (NISTIR 7316, §3.5). Meanwhile, NIST defines ABAC itself as evaluating attributes against policy, rules, or relationships (NIST SP 800-162). The curriculum must therefore define the enterprise's intended use of “Rule-BAC” at the outset and avoid pretending that every vendor uses the term identically.

The recommended product is a **75–90 minute branching-video core**, delivered in short chapters, followed by **role-specific practice and a scored capstone simulation**. The capstone should require participants from business, data, HR, security, IT operations, audit, and service-desk perspectives to decide the same requests, identify every applicable constraint, resolve conflicts using the enterprise's documented policy-combining rule, and take the correct operational action. Passing should require both the right outcome and the right rationale; a lucky “deny” is not evidence of comprehension.

This is an instructional-design blueprint derived from standards and authoritative guidance, not the result of a controlled trial comparing video formats. NIST recommends scenario-based activities, practical assessments that resemble work, measurable outcomes, and evaluation at reaction, learning, behavior, and organizational-results levels, but it does not prescribe this exact curriculum or the passing thresholds proposed below (NIST SP 800-50 Rev. 1).

What the five models mean—and where they overlap

The common decision object

Every lesson should use the same access-request notation:

> ***Subject*** *requests an* ***action*** *on an* ***object/resource****, in an* ***environment****, under one or more* ***enterprise policies****.*

That notation is close to NIST's ABAC definition and prevents learners from confusing authentication (“Who are you?”) with authorization (“May this subject perform this action on this resource now?”). NIST defines access control as the decision to permit or deny a subject access to system objects (NIST glossary).

Lens	Primary decision input	Who controls it?	Typical enterprise expression	Characteristic mistake
DAC	Object ownership and an owner-managed grant	Owner or authorized delegate, within enterprise limits	File share, collaboration invite, ACL entry	“I own it, so I can share it with anyone.”
RBAC	Assigned role and role permissions, hierarchy, and constraints	Central administrators/process owners	Payroll approver, clinician, developer, auditor	Treating a job title as sufficient proof of need
ABAC	Subject, object, action, and environment attributes	Centrally governed policy plus authoritative attribute sources	Department, employment status, data tag, device posture, location, time	Assuming a stale or self-asserted attribute is trustworthy
MAC	Centrally assigned sensitivity labels and formal authorizations	Central authority; not the object owner	Classification or regulated-data label and clearance/authorization	Believing owner consent can override the label

Lens	Primary decision input	Who controls it?	Typical enterprise expression	Characteristic mistake
Rule-BAC / RuBAC	Organization-defined Boolean, temporal, network, risk, or workflow rule	Usually central policy authority	“Allow only during shift, from a managed device, after approval”	Treating any conditional statement as a distinct formal model

NIST describes RBAC as associating permitted actions with roles rather than individual identities; role permissions may be inherited and should reflect defined organizational functions (NIST RBAC glossary). NIST's RBAC work also recognizes core, hierarchical, and constrained forms, including separation-of-duty constraints (NIST RBAC project).

ABAC is broader and more dynamic: authorization is determined by evaluating attributes associated with the subject, object, requested operation, and sometimes the environment against policy (NIST SP 800-162). Its strength is fine-grained, context-sensitive control; its risk is that the decision is only as reliable as attribute definition, issuance, validation, update, and revocation. NIST identifies attribute preparation, veracity, security, readiness, and management as assurance concerns (NIST SP 800-205).

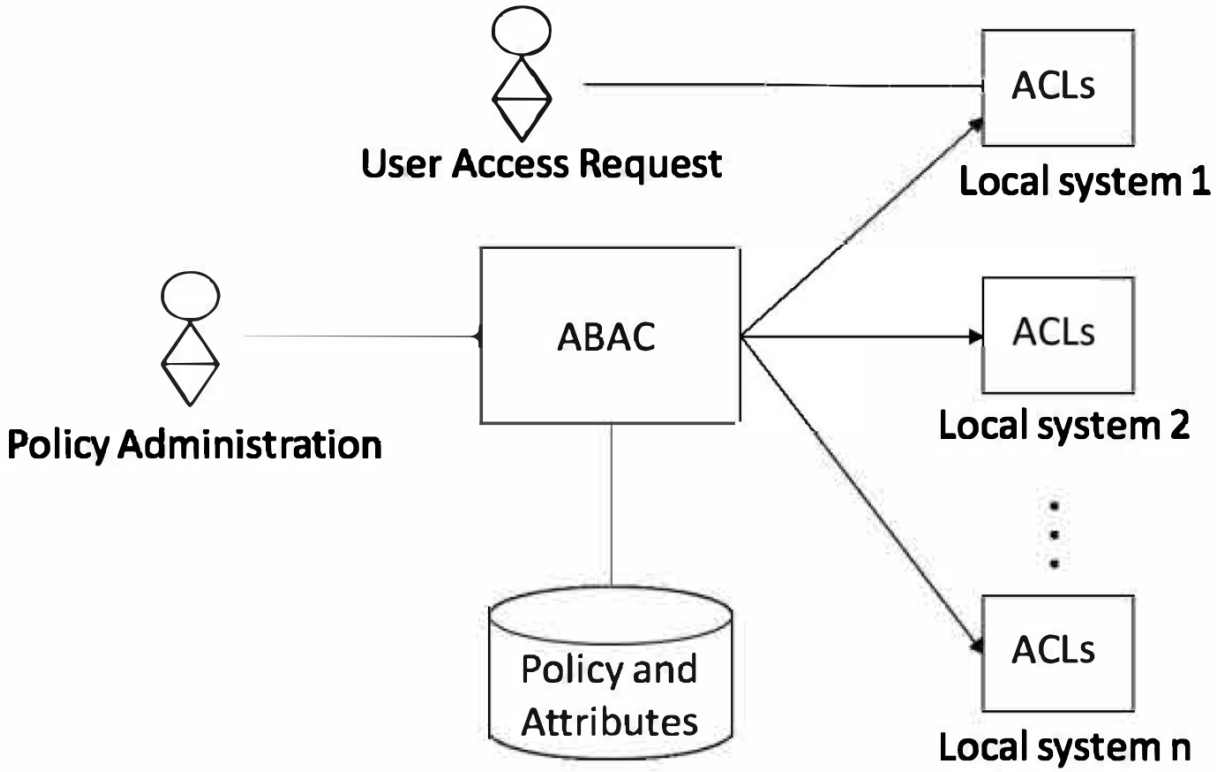

Figure 1. Central ABAC policy administration can coexist with native local ACL enforcement.

Source credit: National Institute of Standards and Technology, “A Method and System for Centralized ABAC Policy Administration and Local Policy Decision and Enforcement Using Access Control Lists” — source

MAC is uniformly enforced and prevents ordinary subjects from changing security attributes, granting their privileges onward, or changing governing rules. A familiar form compares information sensitivity labels with user authorization (NIST MAC glossary). DAC is the converse governance idea: an object owner or authorized controller has discretion to grant rights, pass information, or choose certain security attributes, subject to whatever higher-order constraints the enterprise retains (NIST DAC glossary). NISTIR 7316 cautions that DAC cannot by itself assure information flow after a recipient copies data and that owner decisions may not reflect enterprise-wide requirements (NISTIR 7316, §2.2.1).

The governing mental model: intersection, precedence, and enforcement

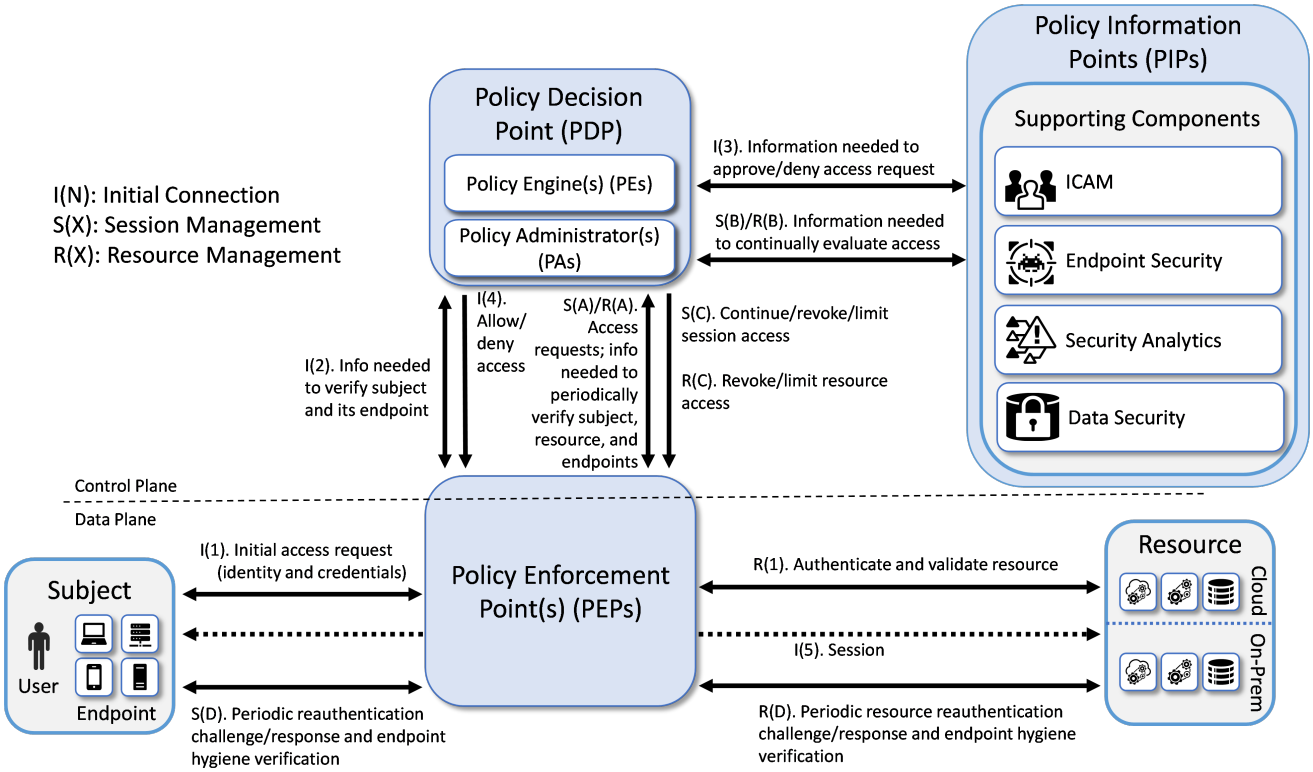

Figure 2. General zero-trust reference architecture: policy information sources inform a distinct decision and enforcement path.

Source credit: National Cybersecurity Center of Excellence, NIST SP 1800-35 supplementary documentation, "Architecture and Builds," Figure 1 — source

The course should animate a single request flowing through several gates:

- **Identity and session facts are established.**
- **RBAC supplies the baseline permissions** appropriate to the work function.
- **ABAC and explicit rules narrow or condition the request** based on current facts.
- **MAC applies non-overridable label/authorization constraints.**
- **DAC may allow the owner to share only inside the remaining boundary.**
- **A policy decision is enforced, logged, and, where appropriate, re-evaluated.**

This sequence is an instructional simplification, not a universal implementation order. Real policy engines may evaluate policies concurrently, and their combining algorithms must be specified. The enterprise should teach one unambiguous local rule—for example, **explicit deny or mandatory constraint overrides a discretionary grant; no applicable permit yields deny; indeterminate results fail closed and escalate**—but label that as enterprise policy rather than a property inherent in all five models.

NIST's zero-trust architecture supplies a useful implementation picture: a policy engine makes the ultimate grant/deny/revoke decision using enterprise policy and supporting information; a policy administrator carries it out; and a policy enforcement point guards the resource. Supporting sources provide identity, device, analytics, and data information (NIST SP 1800-35 architecture). NIST also stresses that network location or asset ownership alone should not create implicit trust and that authentication and authorization occur before a session to a resource (NIST SP 800-207).

Audience and performance outcomes

NIST's NICE approach uses Task, Knowledge, and Skill statements as modular curriculum and assessment building blocks, with work-role assessment typically performed at task level (NICE workforce-framework playbook). Following that method, the program should segment audiences by decisions they make rather than seniority or department name.

Audience tiers

- **All workforce members and contractors**: request, use, share, and report access appropriately.
- **Managers, resource owners, and data stewards**: approve access, classify resources, understand limits of owner discretion, and review exceptions.
- **HR and workforce-identity teams**: maintain authoritative employment, affiliation, training, location, and role attributes through joiner/mover/leaver events.
- **IAM, security, cloud, application, and platform teams**: translate policy into roles, attributes, labels, rules, decision points, and enforcement.
- **Service desk, incident response, privacy, legal/compliance, and audit**: diagnose denials, distinguish defects from correct enforcement, handle emergency access, preserve evidence, and test control effectiveness.

Measurable terminal objectives

After the core and assigned practice, each learner should be able to:

- **Classify** a control statement by its primary lens and explain plausible overlaps.
- **Decompose** a request into subject, action, object, environment, authority, and relevant evidence.
- **Compute** an authorization outcome across at least three simultaneous constraints using the enterprise combining rule.
- **Detect** stale, missing, contradictory, or untrusted attributes and avoid inventing facts.
- **Distinguish** a legitimate DAC grant from a prohibited attempt to override RBAC, ABAC, MAC, or a global rule.
- **Apply** least privilege and separation of duties; NIST defines separation of duties as preventing one user from holding enough privilege to misuse the system alone and notes that it may be static or dynamic (NIST glossary).
- **Choose** the correct operational response: grant, deny, limit, step up authentication, request approval, invoke documented emergency access, revoke, or escalate.
- **Explain and document** the decision in plain language, identifying the decisive policy and evidence source.

Video curriculum blueprint

Design principles

The program should use a life-cycle approach: analyze risks and audiences, design measurable objectives, develop realistic materials, implement them, then assess and improve. That follows NIST's current cybersecurity and privacy learning-program guidance, which emphasizes behavior change, measurement, role-based learning, and continuous improvement (NIST SP 800-50 Rev. 1).

Use a recurring fictional enterprise, “Northstar Group,” rather than five disconnected mini-lectures. Reuse the same people, resources, policies, and audit trail so learners see how a job change, device compromise, label change, or emergency alters the outcome. Present policy cards on screen in exact language; show the attribute source and timestamp; and require a decision before the narrator explains it.

Accessibility is part of design, not post-production: provide accurate captions, transcript, keyboard-operable branches, audio description of meaningful diagrams, non-color status cues, pause/replay controls, and enough time to inspect policy evidence. Local legal and accessibility teams should validate requirements for the delivery jurisdictions.

Core sequence

Chapter	Time	Instructional purpose	Video treatment	Embedded check
0. Diagnostic	5 min	Expose misconceptions without grading	Three fast requests with confidence rating	Capture baseline decision, rationale, and confidence
1. One request, five lenses	8 min	Establish common notation and overlap	Animated policy stack; contrast authentication and authorization	Identify subject, action, object, environment

Chapter	Time	Instructional purpose	Video treatment	Embedded check
2. DAC and its boundary	8 min	Teach ownership, delegation, ACLs, and downstream-copy risk	Resource owner shares a workspace, then meets a regulated-data limit	Select which grants the owner may make
3. RBAC and constraints	10 min	Teach role-permission assignment, hierarchy, least privilege, and SoD	Joiner/mover workflow and payment approval	Detect toxic role combination
4. ABAC and attribute assurance	12 min	Teach subject/object/action/environment attributes and freshness	Remote access changes as device posture and location change	Identify decisive and unreliable attributes
5. MAC and non-override	8 min	Teach labels, formal authorization, and central authority	Executive/resource owner attempts to bypass a label	Explain why authority or ownership is insufficient
6. Rule-BAC without category error	7 min	Teach conditional/global rules and terminology ambiguity	Time, network, training, and risk rules overlay an RBAC grant	Distinguish role fact from conditional rule
7. Combining policies	12 min	Resolve simultaneous permits, denies, missing data, and exceptions	Learner operates a simplified policy-decision console	Calculate outcome and cite decisive constraint
8. Operational response	8 min	Turn a decision into safe action	Service-desk and incident-response handoffs	Choose grant/deny/limit/escalate and audit fields
9. Capstone briefing	5 min	Prepare for simulation and explain scoring	Worked example with think-aloud rationale	No score; calibration only

The diagnostic should not be used to exempt technical staff automatically: familiarity with terms does not prove the ability to resolve interacting policies. Conversely, nontechnical staff do not need policy-language syntax; they need correct decisions, evidence recognition, and escalation behavior.

Role-specific extensions

- **Managers/data owners (20 minutes):** approval evidence, periodic access review, DAC limits, temporary access, conflict of interest, and emergency-access accountability.
- **HR/identity governance (20 minutes):** source-of-truth ownership, joiner/mover/leaver timing, contractor expiration, role assignment evidence, attribute quality, and revocation.
- **Engineers/administrators (35 minutes plus lab):** policy-as-code examples, decision/enforcement separation, deny/indeterminate behavior, test matrices, logging, drift, and rollback.
- **Service desk/IR (20 minutes):** distinguish bad credentials, missing entitlement, stale attribute, correct policy denial, enforcement failure, and suspected compromise; never “fix” a denial by adding a broad role.
- **Audit/privacy/compliance (25 minutes):** traceability from requirement to policy to decision log; sampling; exception evidence; control-owner interviews; and privacy implications of attributes and monitoring.

Scenario-based simulations that test overlapping constraints

Each simulation should disclose only the information available to the learner's work role. Participants may request evidence from simulated systems or colleagues. This tests information gathering and cross-functional handoffs, not just reading comprehension.

1. The traveling payroll approver

Request: A regional finance manager attempts to approve a high-value payroll adjustment from a personal tablet at 02:10 local time while traveling.

Policy stack: RBAC grants payroll-approver permission; a static SoD constraint forbids the same person from initiating and approving; ABAC requires an enterprise-managed, compliant device and current employment status; a global rule restricts high-value approvals to an approved operating window or documented emergency process; DAC is irrelevant because the manager does not own the payroll system.

Branches: The manager initiated the change; device posture is unknown; an executive sends chat approval; the deadline is imminent.

Best response: Deny normal approval, preserve the transaction, invoke only the documented emergency/alternate-approver path, and record the failed device/time/SoD conditions. Executive urgency does not manufacture missing attributes or erase SoD.

What it tests: RBAC is necessary but insufficient; authority is not ownership; rule and ABAC conditions can narrow role permission; business continuity must use an approved exception rather than an improvised override.

2. The clinical researcher and the patient record

Request: A physician who also has a research role exports patient data into a team analytics folder.

Policy stack: The clinician role permits treatment access to assigned patients; the researcher role permits access only to an approved de-identified dataset; ABAC compares care relationship, study approval, purpose, patient-record tag, and export destination; a mandatory privacy/sensitivity label prohibits export of identified records to the research zone; the destination folder owner can invite collaborators under DAC but cannot remove the source label or authorize a prohibited flow.

Branches: The patient is assigned to the physician; the study is approved; the requested export still contains direct identifiers; another researcher asks the folder owner for access.

Best response: Permit treatment use in the clinical context, deny the identified export, route the request through the approved de-identification/data-release workflow, and prevent the folder owner from treating collaboration rights as authority over source data.

What it tests: One individual can activate different roles, but purpose and object attributes still matter; a DAC grant cannot defeat a mandatory label; “can view” does not imply “can export.”

3. The contractor whose project changed

Request: A contractor moved from Project Atlas to Project Beacon yesterday but can still read Atlas source repositories and requests Beacon production logs.

Policy stack: RBAC should remove the Atlas developer role and add a limited Beacon support role after approval; ABAC uses employer, project, contract end date, training completion, device posture, and resource environment; a Rule-BAC control bars production-log access without current privacy training and an active incident ticket; repository maintainers can add collaborators through DAC only within project policy.

Branches: HR has recorded the move, IAM synchronization failed, the old repository owner says continued access is convenient, and the training attribute is expired.

Best response: Revoke stale Atlas access, deny Beacon production logs until role approval and training/ticket conditions are satisfied, treat the synchronization problem as a control incident, and avoid a broad manual group addition.

What it tests: Attribute source and freshness, mover risk, owner discretion limits, enforcement drift, and the difference between correcting data and bypassing policy.

4. The merger data room

Request: Legal creates a merger data room and invites an internal strategy analyst, an external law-firm partner, and a finance employee who is personally connected to the target company.

Policy stack: DAC allows the room owner to invite approved collaborators; RBAC provides legal-deal-team and finance-reviewer capabilities; ABAC checks employer/affiliate, matter assignment, NDA, jurisdiction, conflict status, device, and document tag; a mandatory “deal restricted” label constrains copying and onward sharing; a global ethical-wall rule excludes conflicted personnel.

Branches: The external partner's NDA is valid but their device is unmanaged; the finance employee has the correct general role but a declared conflict; the room owner attempts to forward a file by email.

Best response: Apply the approved external-access path only when all partner conditions are met, deny the conflicted employee despite the finance role, and block onward transfer that violates the mandatory label.

What it tests: External identity, ethical walls, DAC versus mandatory handling, and why generic role membership cannot encode every relationship.

5. Production break-glass during ransomware containment

Request: An incident responder seeks temporary administrative access to isolate production servers while the identity provider reports anomalous sign-ins.

Policy stack: The responder role permits incident actions but not standing domain administration; ABAC requires incident assignment, hardened workstation posture, high-assurance authentication, and current threat state; a Rule-BAC workflow requires two-person approval and a short time-to-live; MAC-like centrally controlled system tiering prevents ordinary admins from changing protected labels; no resource owner can delegate around the emergency controls.

Branches: The primary approver is unreachable; the endpoint becomes noncompliant during the session; isolation risks revenue loss.

Best response: Use the documented alternate approver and break-glass identity, scope and time-limit privilege, record justification, continuously monitor, and revoke or limit the session when posture becomes noncompliant. Conduct post-event review.

What it tests: Just-in-time privilege, dynamic re-evaluation, availability trade-offs, approved exceptions, and revoke—not merely initial grant/deny.

6. The AI assistant and confidential source material

Request: A marketing employee asks an enterprise AI assistant to summarize a confidential engineering document and share the response with an agency.

Policy stack: RBAC allows marketing use of the AI tool but not engineering-source access; ABAC evaluates user department, document tag, approved AI service, tenant, destination, and data-use purpose; a mandatory confidentiality/export label restricts disclosure; the document owner's DAC invitation cannot authorize secondary processing outside policy; a Rule-BAC control blocks prompts containing designated data classes.

Branches: The employee already received a link from the document owner; the AI tool is enterprise-approved; the external agency is under contract but not approved for this project.

Best response: Deny the source use and external disclosure, explain that tool approval and file access do not establish processing purpose or recipient authorization, and route a sanitized-content request to the data owner/engineering and legal review process.

What it tests: Chained actions, purpose limitation, downstream use, recipient attributes, and the false inference that two separate permissions combine into a new permission.

Assessment method and scoring

Evidence captured per decision

For every scored branch, record:

- outcome selected: permit, deny, limit, step-up, revoke, or escalate;
- applicable policy lenses selected;
- decisive facts and their authoritative sources;
- conflict/combining rationale;
- operational follow-up and audit evidence;
- confidence and decision time.

This distinguishes conceptual knowledge from performance. NIST recommends practical assessments aligned to clear objectives and resembling what learners experience in their work roles (NIST SP 800-50 Rev. 1, §5.2.3.3).

Analytic rubric

Dimension	Weight	Full-credit behavior	Critical error
Request decomposition	10%	Correctly identifies subject, action, object, context, and evidence gaps	Confuses authentication with authorization
Constraint coverage	25%	Finds every material role, attribute, rule, label, and owner boundary	Ignores a mandatory/explicit-deny constraint
Decision and precedence	25%	Reaches the policy-correct result and applies combining/indeterminate rule	Permits access prohibited by a non-overridable constraint
Evidence quality	15%	Uses authoritative, current attributes and flags conflicts/staleness	Treats assertion, job title, or urgency as authoritative evidence
Operational response	15%	Scopes, logs, escalates, revokes, or invokes approved exception correctly	Broadens access to make the ticket disappear
Explanation and handoff	10%	Gives concise policy/evidence rationale to the correct function	Cannot identify the decisive policy or owner

Recommended qualification is **80% overall, at least 70% on constraint coverage and decision/precedence, and zero critical unsafe permits**. Technical policy authors and privileged administrators should meet a higher practical threshold—such as 90% with all critical cases correct—before receiving production policy authority. These thresholds are proposed governance choices, not NIST-mandated values; pilot data should be used to calibrate difficulty and cut scores.

Use two forms with equivalent policy structures but different surface details. Randomize attribute values and role assignments so participants cannot pass by memorizing stories. For engineering labs, add a policy-test matrix with positive, negative, boundary, missing-attribute, stale-attribute, conflicting-policy, and revocation cases. NIST research on ABAC testing notes that assurance requires comparing implementation behavior with specified policy across attribute combinations and uses combinatorial methods to make testing tractable (NIST presentation/paper).

Cross-functional team exercise

After individual qualification, run a 45–60 minute tabletop in teams of five to seven. Give HR the identity record, the data steward the classification and purpose rules, IAM the role map, security the device/threat telemetry, the manager the business objective, and audit the decision log. No one receives the whole picture. Score both the final decision and whether the team:

- asks the right function for missing evidence;
- resolves contradictory attributes without silently choosing the convenient value;
- separates policy ownership from technical enforcement;
- uses a documented exception owner;
- preserves an audit trail; and
- communicates an actionable, minimally revealing explanation to the requester.

This evaluates the real enterprise failure mode: not merely one employee misunderstanding a term, but several functions each holding a partial truth and assuming another function checked the rest.

Measurement, validation, and maintenance

Use NIST's four-level evaluation structure:

- **Reaction:** perceived relevance, clarity, accessibility, and cognitive load. Useful for improving delivery, not proof of competence.
- **Learning:** diagnostic-to-posttest change, rubric dimension scores, unsafe-permit rate, rationale quality, and delayed retention at 30–60 days.
- **Behavior:** access-request quality, owner-grant reversals, stale-access remediation, correct escalation, break-glass practice, and supervisor observation.
- **Results:** policy-violation trends, toxic-access exposure time, unauthorized grant rate, audit findings, and incident impact—interpreted cautiously because many factors other than training influence them.

NIST SP 800-50 Rev. 1 describes these as reaction, learning, behavior, and results and recommends multiple feedback sources and continuous improvement (NIST SP 800-50 Rev. 1, §5.2.3). Completion rate alone is therefore an implementation measure, not evidence that employees can resolve overlapping constraints.

Before launch, conduct:

- **Policy-owner validation:** security, privacy, HR, legal/compliance, data governance, and system owners confirm every simulated rule and exception.
- **Cognitive interviews:** representatives explain what each prompt means and why they chose an answer.
- **Accessibility and usability testing:** include assistive-technology users and low-bandwidth delivery.
- **Pilot psychometrics:** inspect item difficulty, discrimination, distractor performance, time, and inter-rater consistency for free-text rationales.
- **Operational correlation:** compare simulation error patterns with access reviews, tickets, and incident lessons without claiming causality.

Refresh scenarios whenever policy, regulation, system architecture, authoritative attribute sources, or exception paths change. Review high-risk branches at least annually and after relevant incidents. Version the course, policy cards, answer key, and source mapping together; retire questions whose “correct” answer no longer matches production policy.

Limitations and practical meaning

The authoritative sources establish access-control concepts and learning-program practices; they do not establish that one 75–90 minute branching video is universally optimal. Duration, thresholds, scenario complexity, and refresh intervals should be piloted. NISTIR 7316 is useful for taxonomy and the ambiguity of Rule-BAC, but it dates from 2006; current enterprise implementation should also use the later ABAC, zero-trust, and learning-program publications cited here.

Terminology remains a material limitation. “Rule-based security policy” can be used differently across sources and products, while ABAC policies necessarily contain rules. The safe practice is to publish a local glossary and always name the actual decision inputs and authority: role, attribute, label, ownership grant, global rule, relationship, or workflow constraint.

Finally, training cannot repair defective authorization design. If roles are overbroad, labels are absent, attributes are stale, exception owners are unclear, enforcement is inconsistent, or logs cannot reconstruct decisions, employees will be asked to reason correctly inside a system that cannot execute the answer. The curriculum should therefore generate a feedback backlog for IAM and policy owners, not conceal recurring design problems as “user error.”

For enterprise leaders, the practical goal is not that every employee recite five definitions. It is that a requester, owner, approver, administrator, responder, and auditor can all look at the same access event and correctly answer: **What evidence applies, which constraint governs, who has authority, what action is safe, and how can we prove it?**

Sources

Claim supported	Evidence type	Source	Direct URL
Current life-cycle guidance for cybersecurity/privacy learning; measurable objectives; scenario-based activities; practical work-like assessment; four evaluation levels	Federal guidance, final special publication	NIST SP 800-50 Rev. 1, *Building a Cybersecurity and Privacy Learning Program* (2024)	https://doi.org/10.6028/NIST.SP.800-50r1
Rule-BAC lacks a commonly accepted formal definition; rules may complement RBAC; DAC, MAC, RBAC characteristics and limitations	Federal technical report	NISTIR 7316, *Assessment of Access Control Systems* (2006)	https://doi.org/10.6028/NIST.IR.7316
ABAC evaluates subject, object, operation, and environment attributes against policy, rules, or relationships	Federal guidance, final special publication	NIST SP 800-162, *Guide to Attribute Based Access Control Definition and Considerations* (updated 2019)	https://doi.org/10.6028/NIST.SP.800-162

Claim supported	Evidence type	Source	Direct URL
Attribute assurance depends on governance, validation, update, revocation, and related lifecycle controls	Federal guidance, final special publication	NIST SP 800-205, *Attribute Considerations for Access Control Systems* (2019)	https://doi.org/10.6028/NIST.SP.800-205
RBAC associates permissions with roles and supports hierarchies and constraints	Federal terminology/project record	NIST CSRC RBAC glossary and project	https://csrc.nist.gov/glossary/term/role_based_access_control; https://csrc.nist.gov/Projects/Role-Based-Access-Control
MAC is centrally/non-discretionarily enforced and constrains onward grant or attribute/rule changes	Federal terminology database drawing from NIST/CNSS publications	NIST CSRC MAC glossary	https://csrc.nist.gov/glossary/term/mandatory_access_control
DAC permits owner/authorized-controller discretion and onward grant, within higher policy limits	Federal terminology database drawing from NIST/CNSS publications	NIST CSRC DAC glossary	https://csrc.nist.gov/glossary/term/discretionary_access_control
Separation of duties prevents one user from holding enough privilege to misuse a system alone and may be static or dynamic	Federal terminology database	NIST CSRC Separation of Duty glossary	https://csrc.nist.gov/glossary/term/Separation_of_Duty
A modular Task, Knowledge, and Skill approach can align curriculum and task-level assessment with work	Federal workforce-framework guidance	NIST NICE, *Playbook for Workforce Frameworks*	https://www.nist.gov/itl/applied-cybersecurity/nice/nice-framework-resource-center/playbook-workforce-frameworks
Zero trust removes implicit trust based solely on location/ownership and separates policy decision, administration, enforcement, and supporting information	Federal final special publications and implementation guide	NIST SP 800-207 and NIST SP 1800-35 supplementary architecture	https://doi.org/10.6028/NIST.SP.800-207; https://pages.nist.gov/zero-trust-architecture/VolumeB/architecture.html
ABAC implementations require systematic tests over combinations of attributes; combinatorial methods can reduce the test set	NIST-hosted research presentation/paper	Kuhn et al., *Pseudo-exhaustive Testing of Attribute Based Access Control Rules*	https://csrc.nist.gov/CSRC/media/Presentations/Pseudo-exhaustive-Testing-of-Attribute-Based-Acces/images-media/abac-pseudo-ex-iwct.pdf
Central ABAC administration can be translated into and enforced through local ACL mechanisms; implementation trade-offs exist	NIST patent/technology description	NIST, *A Method and System for Centralized ABAC Policy Administration and Local Policy Decision and Enforcement Using Access Control Lists*	https://www.nist.gov/patents/method-and-system-centralized-abac-policy-administration-and-local-policy-decision-and