InCommon BaseCAMP
JUNE 1–5, 2026

*x*BAC: An Overview of Access Control

Kellen Murphy, Identity Architect | University of Virginia

Date: June 2, 2026

About Me

Kellen Murphy

- Identity Architect @ University of Virginia ~4 yrs
- Internet2/InCommon Technical Advisory Committee
- BaseCAMP Planning Committee
- Before UVA – IAM Consultant ~7 yrs

What I work on daily

- Grouper, Shibboleth, Keycloak, Delinea/PAM, Fischer IGA (if I must)
- Making access control decisions that don't haunt us later 🤔

Last year at BaseCAMP

- Core Concepts of Access & Grouping (BaseCamp 2025)
- This session builds on that session…

Session Overview

XBAC: An Overview of Access Control

Part 1
Access Control Methods: RBAC, ABAC, PBAC, ReBAC

Part 2
Groups as the Bridge

Part 3
Higher Education Context & Challenges

Part 4
The Bigger IAM Picture: Federation, Governance, & Putting It All Together

Setting the Stage

A quick context check before we dive in

Authentication – *AuthN*

"Who are you?"

- Validation of identity claims
- Single-factor (SFA) or Multi-factor (MFA)
- Happens at the Identity Provider (IdP)
- SAML assertions, OIDC tokens

Authorization – *AuthZ*

"What can you do?"

- Can happen at IdP or Service Provider (SP) / Relying Party (RP)
- Requires attributes, roles, or policies… **identity data**!
- SAML attributes, OIDC claims, REST API calls
- THIS is what access control is all about
- The rest of this talk lives here

Part 1

Access Control Methods

What Is Access Control?

Part 1: Access Control Methods

Access control is the practice of restricting who can perform what actions on which resources.

Three classic questions:

Who?
Subject — the user, service, or system requesting access

What?
Action — read, write, execute, approve, delete…

Where?
Object / Resource — the file, system, application, or data

The “model” you choose determines how you answer these questions at scale.

RBAC: Role-Based Access Control

Part 1: Access Control Methods

Role-Based Access Control

Access is granted based on a user's role within the organization.

Benefits

- Simple mental model
- Easy to audit
- Scales well in stable orgs
- Familiar to admins

Challenges

- Role explosion as org grows
- Coarse-grained by nature
- Doesn't handle context well
- Static — ignores runtime conditions

Classic Example

Roles:

- student
- faculty
- staff
- admin

A user has a role.
A role has permissions.
The user inherits those permissions.

It's simple — until you need to deal with…

"faculty who are also students"

"staff with research access only during project Y."

RBAC in Practice: PeopleSoft SIS

Part 1: Access Control Methods

Scenario: New Registrar Hire

A new Registrar's Office staffer needs to process enrollments in PeopleSoft (the SIS).

How It Works

- HR job code → role: records-specialist
- Grouper builds the role as a group
- Shibboleth releases it; PeopleSoft enforces

Why RBAC Fits

- Stable & coarse-grained — easy to audit
- Auto-deprovisions on job change
- All in TAP — no extra infrastructure

TAP Components

Grouper

Role modeled as a group
Membership from HR job data

Shibboleth IdP

Releases the role as a SAML attribute (isMemberOf) to PeopleSoft

The Seam → ABAC

"records-specialist — **only** for the College of Arts & Sciences"

Pure RBAC → records-specialist-AS, -Eng, -Med…
a.k.a. role explosion

ABAC: Attribute-Based Access Control

Part 1: Access Control Methods

Attribute-Based Access Control

Access decisions are based on attributes of the subject, resource, action, and environment.

Benefits

- Fine-grained, contextual
- Flexible policy expression
- Handles complex conditions
- No role proliferation

Challenges

- Complex rules — hard to audit
- Data completeness is critical
- Requires reliable attribute sources
- Can become opaque quickly

Example

ALLOW access IF:

- subject.affiliation = “employee”
- subject.department = “research”
- resource.sensitivity = “low”
- environment.time = business_hours

Attribute types:

- Subject: who the user is
- Resource: what is being accessed
- Action: what they want to do (and why)
- Environment: when, from where, how

ABAC in Practice: Restricted Research Data

Part 1: Access Control Methods

Scenario: HIPAA Research Dataset

A grad researcher needs a dataset containing protected health information (PHI).

The Policy — ALLOW if:

- Affiliation = active member/employee
- isMemberOf = study:IRB-2026-114:approved
- hipaaTrainingCurrent = true
- Request from a managed device

Why ABAC Fits

- No role explosion — one policy, all studies
- Context-aware: training, device, time
- Fine-grained without new roles
- Only as good as attribute freshness

What Carries It

Shibboleth IdP

- Releases affiliation + isMemberOf
- Releases the training entitlement

Grouper

- Computes the IRB-approval group

Beyond TAP

- Decision logic lives within the app/RP
- Device posture → MDM; training → LMS

The Seam → PBAC

- Many rules across apps → centralize & audit

PBAC: Policy-Based Access Control

Part 1: Access Control Methods

Policy-Based Access Control

Access is governed by explicit, centrally managed policies — often combining role and attribute logic.

Benefits

- Centralized policy management
- Separates policy from enforcement
- Auditable and explainable
- Supports compliance requirements

Challenges

- Requires dedicated infrastructure
- Policy language complexity (XACML, OPA)
- Centralized enforcement can be a bottleneck
- Often $$$

Key Concepts

PDP
Policy Decision Point — evaluates the policy

PEP
Policy Enforcement Point — enforces the decision

PAP
Policy Administration Point — manages policies

Examples:

- AWS IAM Policies
- OPA (Open Policy Agent)
- XACML-based systems
- PAM policy engines (CyberArk, Delinea)

PBAC in Practice: FERPA Records

Part 1: Access Control Methods

Scenario: FERPA Across Systems

The rule for who may view student records must hold identically in the SIS, advising, degree audit, warehouse, and dashboards.

The Policy — ALLOW read if:

- Active employee/staff affiliation
- Legitimate Educational Interest (LEI) entitlement
- FERPA training current
- Not caught by an SoD deny

Why PBAC Fits

- One policy, enforced identically everywhere
- Auditable via decision logs
- Compliance-friendly (FERPA)
- Watch-out: PDP infra, single point of failure, learning curve

What Carries It

Shibboleth IdP

Releases affiliation, isMemberOf, training entitlement

Grouper

Computes LEI populations + SoD deny groups

Beyond TAP

OPA = central PDP/PAP; apps = PEPs; logs → audit

The Seam → ReBAC

- Access tied to a specific relationship (advisor ↔ advisee) → ReBAC

ReBAC: Relationship-Based Access Control

Part 1: Access Control Methods

Relationship-Based Access Control

Access decisions are based on the relationship between the subject and the resource (or other entities).

Benefits

- Models real-world "ownership" naturally
- Great for hierarchical or graph-like resources
- Context-aware without attribute explosion
- Powers Google Zanzibar, AWS Cedar

Challenges

- Requires relationship graph infrastructure
- Newer — fewer mature tools in higher ed
- Can be hard to explain to stakeholders
- Needs careful graph design

Example: Document Access

ALLOW IF:

- user IS owner of document
- user IS member of document's group
- user IS editor of document's parent folder

Real-world implementations:

- Google Drive sharing model
- GitHub org/repo permissions
- AWS Verified Permissions (Cedar)
- OpenFGA (open-source Zanzibar)

Higher ed relevance: research collaboration, data governance, shared resource ownership

ReBAC in Practice: Advisor ↔ Advisee

Part 1: Access Control Methods

Scenario: Advisor ↔ Advisee

An advisor should see records for their assigned advisees — not the entire student body.

The Relationships

- Advisor —[advises]→ student
- Student —[member_of]→ program
- Advisor —[directs]→ program (transitive)
- Check: does an "advises" path exist?

Why ReBAC Fits

- Models the real advisor↔advisee tie
- No per-advisor group explosion
- Handles transitivity (program → cohort)
- Watch-out: newest, needs a graph store

What Carries It

Shibboleth IdP

- Authenticates; supplies identity (eppn)

Grouper

- Seeds edges, but per-advisor = explosion 💥

Beyond TAP

- Relationship graph + check API
- OpenFGA / SpiceDB ("Zanzibar")
- AWS Cedar, Okta FGA
- Azure: no equivalent (RBAC/ABAC) 😏

The Reality → They Work Together

Real systems combine RBAC + ABAC + PBAC + ReBAC

Comparison: The Four Approaches

Part 1: Access Control Methods

RBAC · Roles
✓ Simple, auditable ⚠️ Role explosion, coarse-grained

ABAC · Attributes
✓ Fine-grained, flexible ⚠️ Data completeness, opaque rules

PBAC · Policies
✓ Centralized, compliant ⚠️ Infrastructure cost, complexity

ReBAC · Relationships
✓ Ownership-aware, dynamic ⚠️ Newer tooling, graph overhead

Takeaway: *x*BACs Work Together

Part 1: Access Control Methods

The Real World

No single model fits every use case.

Most organizations combine approaches:

- RBAC for broad, stable permissions
- ABAC to refine based on attributes
- PBAC to formalize compliance requirements
- ReBAC for ownership / collaboration scenarios

The goal: access decisions that are correct, explainable, and maintainable.

Higher Ed Example:

Canvas LMS access

- RBAC: instructor / student / TA roles
- ABAC: active enrollment attribute
- PBAC: policy: “must be enrolled AND active”
- ReBAC: TA can view only sections they are assigned to

These aren’t competing models — they solve different dimensions of the same problem.

Part 2

Groups as the Bridge

Where Groups Fit In

Part 2: Groups as the Bridge

Groups bridge the gap

Groups are how people get into access control models — they are the mechanism that connects identity to authorization.

Groups as RBAC roles:

- A group IS a role
- Membership = role assignment
- Groups in Grouper can be SAML attributes

Groups as ABAC attributes:

- Group membership surfaced as user attribute
- “member of research-data-approved”
- Passed in SAML assertion or OIDC claim

The chain of trust:

HR System / SIS
Authoritative source of role/affiliation

Grouper / IGA
Group logic, composition, provisioning

LDAP / Directory
Group stored as attribute

IdP (Shibboleth)
Releases group as SAML attribute

Service Provider
Makes access decision from attribute

The group is the signal. The access control model is the interpreter.

Group Data Structures

Part 2: Groups as the Bridge

Flat Groups

Direct membership only. Common in LDAP deployments.

- Simple to understand
- Easy to query
- Manual membership — entropy risk
    - Sprawl is real!
- No hierarchy or composition

Nested Groups

Groups within groups. Indirect membership.

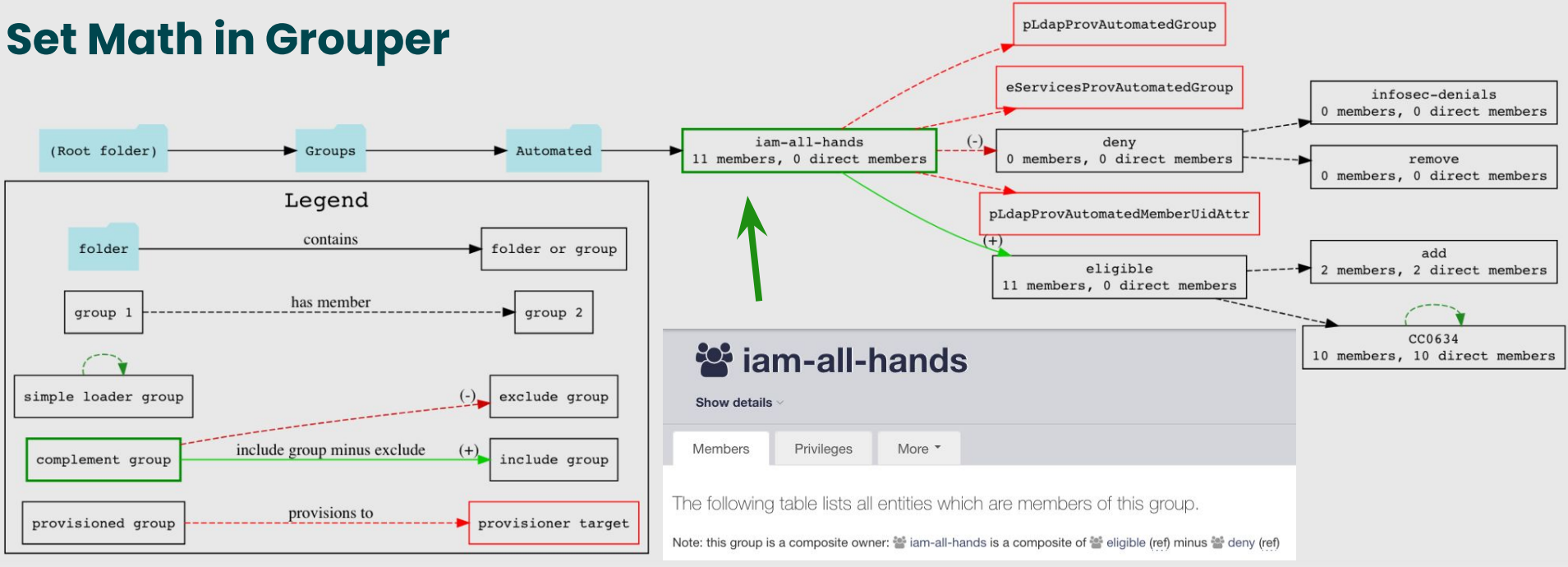
- Hierarchical — reflects org structure
- Indirect members inherit from parent
- Powerful for role delegation
- Can be complex to debug
- Can be challenging to maintain

There is no one right answer!

Group Data Automation

Part 2: Groups as the Bridge

Set Math in Grouper



Part 3

Higher Education Context

Higher Ed: Unique Challenges

Part 3: Higher Education Context

Multi-Role Populations

- Faculty who are also students (grad programs, dual appointments)
- Staff who teach, students who work in IT, external collaborators
- Traditional RBAC struggles with people who wear many hats

Institutional Complexity

- Decentralized IT — departments manage their own systems
- Legacy systems with no attribute awareness
- Research computing needs that differ from admin needs

Federated Identity Constraints

- SAML attribute release policies govern what IdPs send
- InCommon metadata and attribute bundles shape what SPs receive
- Cross-institutional collaboration adds another layer (CILogon, EduGAIN)

The Role Explosion Problem

Part 3: Higher Education Context

The Role Explosion Problem

Pure RBAC creates a combinatorial nightmare in higher ed.

Example: access to a research dataset

- faculty-biology-read
- faculty-biology-write
- grad-biology-read
- staff-research-read
- postdoc-biology-read
- … and it keeps growing

With 200 departments × 5 resource types × 3 permission levels… that's a lot of roles.

Better Approaches

Use ABAC for context

Add affiliation + department as attributes, reduce roles needed

Composite groups

Reference groups + set math to derive population without creating roles

Policy-based refinement

One role + PBAC policy to scope access to specific resources

Governance checkpoints

Attestation and certification to catch role sprawl early

The goal: meaningful roles that survive org changes, not roles that replicate your org chart.

Part 4

The Bigger IAM Picture

The IAM Conceptual Model

Part 4: The Bigger IAM Picture

Key capability areas relevant to access control:

Identity Governance
Who has access? Is it appropriate? Attestation, certification, SoD

Access Management
SSO, MFA, federation — AuthN before AuthZ

Provisioning / IGA
How do users get access? Groups, roles, entitlements

Directory & Data Services
The attribute store — groups and roles live here

Access control sits at the intersection of ALL four. Your access model only works when the other capabilities are reliable.

The IAM Conceptual Model

Part 4: The Bigger IAM Picture

Key capability areas relevant to access control:

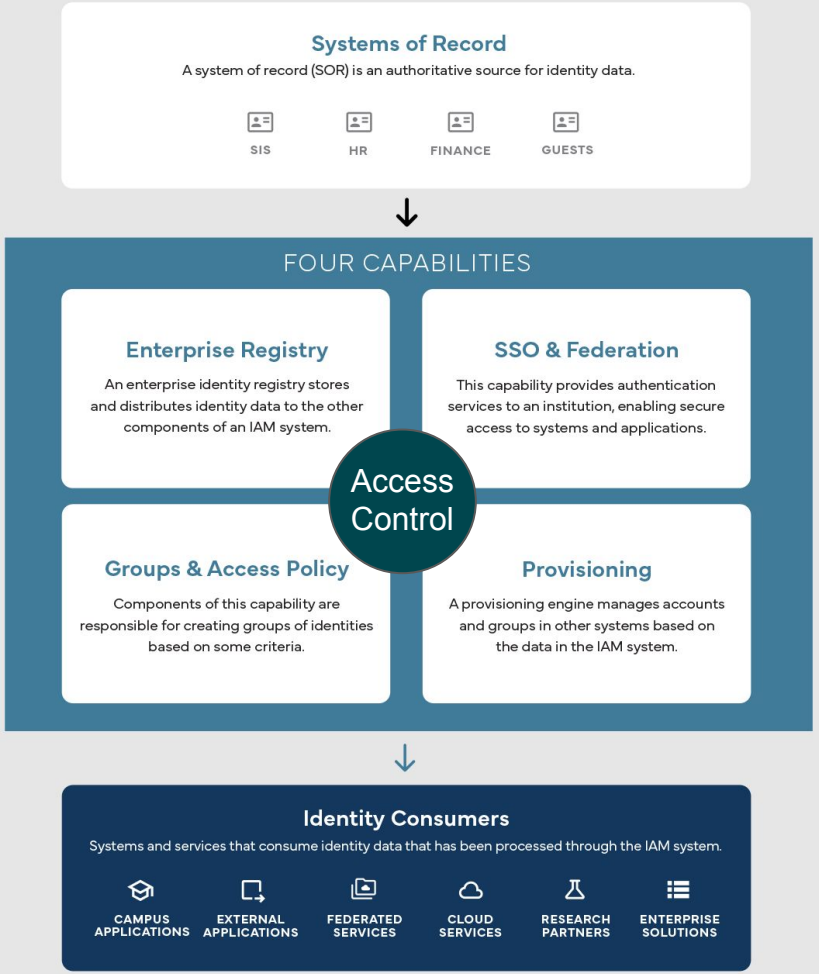
Identity Governance
Who has access? Is it appropriate? Attestation, certification, SoD

Access Management
SSO, MFA, federation — AuthN before AuthZ

Provisioning / IGA
How do users get access? Groups, roles, entitlements

Directory & Data Services
The attribute store — groups and roles live here

Access control sits at the intersection of ALL four. Your access model only works when the other capabilities are reliable.



Access Control + Federated Authentication

Part 4: The Bigger IAM Picture

Federation as AuthN Infrastructure

Federated identity (SAML, OIDC) is the pipeline that carries your AuthZ signals to SPs.

The IdP is a policy enforcement point:

- Releases attributes to SPs (roles, groups, affiliation)
- Attribute release policy = authorization policy
- SAML eppn, eduPersonAffiliation, isMemberOf
- InCommon attribute bundles standardize this

SIRTFI and incident response:

- Access revocation must propagate across federation
- IdP must be able to disable access quickly
- This is an access control responsibility too

The Attribute Pipeline

SIS/HR
Authoritative affiliation, enrollment, department

Grouper / IGA
Group membership derived from SIS + manual overrides

LDAP
Groups and attributes stored and queryable

Shibboleth IdP
Attributes resolved + released per SP policy

SP / App
Makes access decision: role or attribute match

Garbage in, garbage out. If your source data is wrong, your access policy is wrong.

Governance & Attestation

Part 4: The Bigger IAM Picture

Access governance closes the loop

Giving access is easy. Maintaining appropriate access over time is hard.

Attestation

- Group / role owners periodically certify that membership is correct
- Triggered by time, events (offboarding), or compliance requirements
- What happens on failure? Deprovision? Notify? Escalate?

Access Certification

- Broader reviews of who has access to what (often for compliance — HIPAA, PCI, SOX)
- Driven by risk — high-sensitivity resources reviewed more often
- Requires reliable access model to be meaningful

Segregation/Separation of Duties (SoD)

- No single person should have incompatible privileges (e.g. create + approve invoices)
- Relevant in higher ed for financial systems, student records, HR
- RBAC makes this easier to model — ABAC/PBAC make it harder without tooling

Putting It All Together

Part 4: The Bigger IAM Picture

A defensible access control model in higher ed looks like this:

1. Authoritative sources
SIS and HR feed group membership automatically

2. Group logic
Grouper (or equivalent) handles composition, exceptions, and set math

3. Access model
RBAC for broad roles + ABAC attributes refine + PBAC centralizes policy

4. Federation
IdP releases the right attributes to the right SPs — no over-sharing

5. Governance
Attestation and certification verify that access stays appropriate over time

These aren't separate systems — they're interconnected. Access control decisions are only as good as the data and governance behind them.

InCommon BaseCAMP
JUNE 1-5, 2026

Thanks!
Any Questions?

Feel free to email me (kellen.murphy@virginia.edu)
or find me on the Internet2 Slack (@KellenMurphy)