Solution Architecture - Optimizing User, Group, and Role Management with Access Control & Automated Workflows

Architecture Objective

To design a scalable and secure IAM ecosystem integrating cloud-native microservices, automated workflows, and role-based access controls (RBAC/ABAC).

Core Components

Identity Provider (IDP), Role Engine, Workflow Automation Layer, Access Policy Engine, Audit and Monitoring Subsystem.

Enterprise Context

Designed for hybrid environments supporting on-prem directories and multi-cloud deployments with federated authentication (SAML/OAuth2).

Security Design

Implements Zero Trust architecture principles with continuous verification, least privilege enforcement, and adaptive risk-based authentication.

Integration Layer

REST APIs connect frontend request handlers with backend IAM services ensuring stateless scalable transactions.

Diagram 1: Enterprise IAM Architecture (C-Ent)

USERS → IDENTITY PROVIDER (SSO/SAML) → WORKFLOW ENGINE → ROLE ENGINE → POLICY ENFORCEMENT POINT → MONITORING DASHBOARD

Diagram 2: Workflow Automation Stack

Frontend Portal → Access Request API → Workflow Queue → Rule Engine → Database → Audit Logs

Layer	Technology Stack	Function
Identity Layer	OAuth2, SAML, OpenID Connect	Federated login & verification
Access Layer	RBAC/ABAC Engine, Policy Scripts	Access evaluation & decision
Workflow Layer	Python Workflow Engine, REST API	Request routing & automation
Audit Layer	Elastic Stack, SIEM	Continuous monitoring

Diagram 3: Multi-Cloud IAM Enterprise Integration (C-Ent)

ON-PREM AD → FEDERATION GATEWAY → CLOUD IAM SERVICES (AWS IAM / Azure AD / GCP IAM) → CENTRAL POLICY HUB

Each cloud node communicates via API gateway and policy sync layer, ensuring unified governance and centralized privilege revocation capability.

Conclusion

The enterprise-grade architecture delivers automation, compliance, and scalability through interconnected IAM layers, enabling efficient user, group, and role management workflows.