AYI NEDJIMI Consultants

Cloud IAM : Sécurisation des Identités et Accès AWS,

Catégorie : Cloud Security Lecture : 5 min Publié le : 08/03/2026 Auteur : Ayi NEDJIMI

Guide complet Cloud IAM : sécurisation des identités et accès sur AWS, Azure et GCP. Politiques least privilege, attaques IAM, PIM, CIEM, IRSA.

```
# Workload Identity Federation - GitHub Actions vers GCP
# 1. Créer un Workload Identity Pool
gcloud iam workload-identity-pools create "github-pool" \
  --project="my-project" \
  --location="global" \
  --display-name="GitHub Actions Pool"

# 2. Créer un provider OIDC pour GitHub
gcloud iam workload-identity-pools providers create-oidc "github-provider" \
  --project="my-project" \
  --location="global" \
  --workload-identity-pool="github-pool" \
  --display-name="GitHub Provider" \
  --attribute-
mapping="google.subject=assertion.sub,attribute.repository=assertion.repository" \
  --issuer-uri="https://token.actions.githubusercontent.com"

# 3. Autoriser le workflow GitHub à impersonner un service account
gcloud iam service-accounts add-iam-policy-binding \
  deploy-sa@my-project.iam.gserviceaccount.com \
  --project="my-project" \
  --role="roles/iam.workloadIdentityUser" \
  --member="principalSet://iam.googleapis.com/projects/123456/locations/global/
workloadIdentityPools/github-pool/attribute.repository/my-org/my-repo"

# 4. Dans le workflow GitHub Actions :
# - uses: google-github-actions/auth@v2
#   with:
#     workload_identity_provider: 'projects/123456/locations/global/workloadIdentityPools/
github-pool/providers/github-provider'
#     service_account: 'deploy-sa@my-project.iam.gserviceaccount.com'
```

5.4 IAM Conditions et Organisation Policies

GCP offre des **IAM Conditions** pour restreindre les permissions en fonction de critères contextuels (heure, attributs de la ressource, adresse IP). Les **Organization Policies** permettent de définir des contraintes à l'échelle de l'organisation, similaires aux SCP d'AWS : Guide complet Cloud IAM : sécurisation des identités et accès sur AWS, Azure et GCP. Politiques least privilege, attaques IAM, PIM, CIEM, IRSA. La sécurité du cloud requiert une compréhension approfondie des modèles de responsabilité partagée. Ce guide sur cloud iam securite identites acces s'adresse aux architectes et ingénieurs sécurité. Nous abordons notamment : 8. scénarios

d'attaque iam et remédiation, questions frequentes et 9. conclusion : vers une posture iam mature. Les professionnels y trouveront des recommandations actionnables, des commandes prêtes à l'emploi et des stratégies de mise en œuvre adaptées aux environnements d'entreprise.

```
# IAM Condition : autoriser l'accès uniquement pendant les heures ouvrées
gcloud projects add-iam-policy-binding my-project \
  --member="user:admin@example.com" \
  --role="roles/compute.admin" \
  --condition='expression=request.time.getHours("Europe/Paris") >= 8 &&
request.time.getHours("Europe/Paris") <= 18,title=heures-ouvrees,description=Accès limité
aux heures de bureau'

# Organization Policy : interdire les clés de service account
gcloud resource-manager org-policies enable-enforce \
  constraints/iam.disableServiceAccountKeyCreation \
  --organization=123456789

# Organization Policy : restreindre les domaines autorisés
gcloud resource-manager org-policies set-policy policy.yaml \
  --organization=123456789

# policy.yaml :
# constraint: constraints/iam.allowedPolicyMemberDomains
# listPolicy:
#   allowedValues:
#     - "C0xxxxxxx"  # Customer ID de votre organisation Google Workspace

# Organization Policy : restreindre les régions de déploiement
gcloud resource-manager org-policies set-policy region-policy.yaml \
  --organization=123456789
# constraint: constraints/gcp.resourceLocations
# listPolicy:
#   allowedValues:
#     - "europe-west1"
#     - "europe-west9"  # Paris
```

7.1 Principe du moindre privilège en pratique

Le **principe du moindre privilège (PoLP)** est le fondement de toute stratégie IAM, mais son application pratique reste le défi principal. Voici les techniques concrètes pour implémenter le PoLP sur les trois clouds :

```
# === AWS : Utiliser IAM Access Analyzer pour identifier les permissions inutilisées ===
# Générer une politique basée sur l'activité réelle (last accessed)
aws accessanalyzer generate-policy \
  --policy-generation-details '{"principalArn":"arn:aws:iam::123456789:role/my-app-role"}'
\
  --cloud-trail-details '{"trails":[{"cloudTrailArn":"arn:aws:cloudtrail:eu-
west-1:123456789:trail/main","regions":["eu-
west-1"],"allRegions":false}],"accessRole":"arn:aws:iam::123456789:role/
AccessAnalyzerRole","startTime":"2025-01-01T00:00:00Z","endTime":"2025-03-01T00:00:00Z"}'

# Vérifier les permissions inutilisées (last accessed)
aws iam generate-service-last-accessed-details \
  --arn arn:aws:iam::123456789:role/my-app-role

# === Azure : Utiliser Entra ID Access Reviews ===
# Créer une Access Review pour les rôles privilégiés
az rest --method POST \
  --url "https://graph.microsoft.com/v1.0/identityGovernance/accessReviews/definitions" \
  --body '{
    "displayName": "Revue trimestrielle - Global Administrators",
    "scope": {
      "query": "/roleManagement/directory/roleAssignments?$filter=roleDefinitionId eq
'\''62e90394-69f5-4237-9190-012177145e10'\''",
      "queryType": "MicrosoftGraph"
    },
    "reviewers": [{"query": "/users/security-
manager@contoso.com","queryType":"MicrosoftGraph"}],
    "settings": {
      "mailNotificationsEnabled": true,
      "autoApplyDecisionsEnabled": true,
      "defaultDecision": "Deny",
      "recurrence": {"pattern":{"type":"absoluteMonthly","interval":3}}
    }
  }'

# === GCP : Utiliser Policy Analyzer et Recommender ===
# Analyser les permissions effectivement utilisées
gcloud policy-intelligence query-activity \
  --project=my-project \
  --activity-type=serviceAccountLastAuthentication

# Obtenir les recommandations de réduction de permissions
gcloud recommender recommendations list \
  --project=my-project \
  --location=global \
  --recommender=google.iam.policy.Recommender
```

7.2 Gestion des credentials et rotation

La gestion des credentials est le talon d'Achille de la sécurité IAM. Les access keys, client secrets et service account keys sont des vecteurs de compromission persistants qui survivent au-delà de la session de l'utilisateur. Les bonnes pratiques universelles :

- **Privilégier les identités fédérées** : SSO via SAML/OIDC plutôt que des utilisateurs IAM locaux
- **Utiliser les identités machine natives** : IRSA (AWS), Managed Identities (Azure), Workload Identity (GCP) plutôt que des clés statiques

- **Rotation automatique** : max 90 jours pour les clés d'accès, 180 jours pour les client secrets avec alerte 30 jours avant expiration
- **Détection des credentials exposés** : GitHub secret scanning, AWS Secrets Manager rotation, Azure Key Vault avec expiration automatique
- **MFA obligatoire** : pour tous les comptes humains, avec phishing-resistant (FIDO2/passkeys) pour les comptes privilégiés

Credentials dans le code : le risque n-1
Selon le rapport GitGuardian 2025, plus de **12 millions de secrets** ont été exposés dans des repositories publics en 2024, dont 35 % sont des credentials cloud (AWS access keys, Azure client secrets, GCP service account keys). Un scan automatisé avec des outils comme `trufflehog`, `gitleaks` ou `detect-secrets` doit faire partie du pipeline CI/CD. Toute clé exposée doit être considérée comme compromise et immédiatement révoquée.

7.3 Monitoring et détection des anomalies IAM

Le monitoring IAM doit détecter trois catégories de menaces : les **accès non autorisés** (credentials volés ou fédération compromise), l'**escalade de privilèges** (modification de politiques ou de rôles) et les **mouvements latéraux** (utilisation de permissions cross-account ou cross-project). Les événements critiques à surveiller :

```
# === AWS CloudTrail - Événements IAM critiques ===
# Détection : création d'un utilisateur IAM avec des clés d'accès
aws cloudtrail lookup-events \
  --lookup-attributes AttributeKey=EventName,AttributeValue=CreateAccessKey \
  --start-time "2025-03-01T00:00:00Z" \
  --end-time "2025-03-08T00:00:00Z"

# Événements critiques à alerter :
# - CreateUser, CreateAccessKey, AttachUserPolicy (création de persistence)
# - AssumeRole avec source inhabituelle (mouvement latéral)
# - PutRolePolicy, CreatePolicyVersion (escalade de privilèges)
# - ConsoleLogin sans MFA depuis une IP inconnue
# - DeactivateMFADevice (désactivation de MFA)

# === Azure - Entra ID Audit & Sign-in Logs ===
# Requête KQL dans Log Analytics / Sentinel
# SigninLogs
# | where ResultType == 0  // connexions réussies
# | where RiskLevelAggregated in ("high", "medium")
# | where AppDisplayName in ("Azure Portal", "Microsoft Graph")
# | project TimeGenerated, UserPrincipalName, IPAddress, Location, RiskDetail
# | sort by TimeGenerated desc

# === GCP - Cloud Audit Logs ===
# Détection : modifications IAM suspectes
gcloud logging read \
  'protoPayload.methodName="google.iam.admin.v1.SetIamPolicy" OR
   protoPayload.methodName="google.iam.admin.v1.CreateServiceAccountKey" OR
   protoPayload.methodName="SetIamPolicy"' \
  --project=my-project \
  --freshness=24h \
  --format=json
```

8. Scénarios d'attaque IAM et remédiation

8.1 Scénario 1 : Escalade de privilèges via politique IAM permissive

L'attaque la plus classique consiste à exploiter une politique IAM qui accorde `iam:*` ou des permissions de type `*:*`. Un attaquant disposant d'un accès limité peut créer de nouveaux rôles, s'attribuer des permissions administratives, ou modifier les politiques existantes pour élever ses privilèges :

```
# Attaque : un développeur dispose de iam:CreatePolicyVersion
# Il peut modifier une politique existante pour s'octroyer admin
aws iam create-policy-version \
  --policy-arn arn:aws:iam::123456789:policy/dev-policy \
  --policy-document '{
    "Version":"2012-10-17",
    "Statement":[{
      "Effect":"Allow",
      "Action":"*",
      "Resource":"*"
    }]
  }' \
  --set-as-default

# Remédiation : Permission Boundary qui bloque les actions IAM dangereuses
# + SCP au niveau de l'OU qui interdit iam:CreatePolicyVersion
# sauf pour le rôle d'administration centralisé
```

8.2 Scénario 2 : Persistance via fédération SAML/OIDC

Un attaquant ayant compromis un compte administratif cloud peut créer un **Identity Provider (IdP) fédéré** qui pointe vers son propre fournisseur SAML. Cela lui permet de s'authentifier à tout moment avec des rôles arbitraires, même après la rotation des credentials compromis initialement. Cette technique de persistance est particulièrement dangereuse car elle survit aux resets de mots de passe et à la révocation des clés :

```
# Attaque : création d'un IdP SAML malveillant dans AWS
aws iam create-saml-provider \
  --saml-metadata-document file://evil-metadata.xml \
  --name "BackdoorIdP"

aws iam create-role \
  --role-name "BackdoorRole" \
  --assume-role-policy-document '{
    "Version":"2012-10-17",
    "Statement":[{
      "Effect":"Allow",
      "Principal":{"Federated":"arn:aws:iam::123456789:saml-provider/BackdoorIdP"},
      "Action":"sts:AssumeRoleWithSAML",
      "Condition":{"StringEquals":{"SAML:aud":"https://signin.aws.amazon.com/saml"}}
    }]
  }'

# Remédiation :
# 1. Auditer régulièrement les IdP configurés
aws iam list-saml-providers
aws iam list-open-id-connect-providers

# 2. SCP pour bloquer la création d'IdP non autorisés
# 3. Alerte CloudTrail sur CreateSAMLProvider / CreateOpenIDConnectProvider
# 4. Revue mensuelle des trust relationships de tous les rôles
```

8.3 Scénario 3 : Mouvement latéral cross-cloud

Dans les environnements multi-cloud, un attaquant qui compromet un environment AWS peut pivoter vers Azure ou GCP via des **fédérations inter-cloud** mal configurées. Par exemple, un Workload Identity Federation GCP qui accepte des tokens OIDC d'AWS sans restriction sur le rôle source permet à tout rôle AWS d'accéder aux ressources GCP. La remédiation consiste à restreindre les conditions de fédération au strict minimum (rôle source spécifique, account ID, claims OIDC). Consultez notre article sur les erreurs de configuration cloud pour d'autres scénarios d'attaque.

Pour approfondir ce sujet, consultez notre outil open-source gcp-security-scanner qui facilite l'analyse de sécurité Google Cloud Platform.

Questions frequentes

Comment mettre en place Cloud IAM dans un environnement de production ?

La mise en place de Cloud IAM en production necessite une planification rigoureuse, incluant l'evaluation des prerequis techniques, la definition d'une architecture cible, des tests de validation approfondis et un plan de deploiement progressif avec des points de controle a chaque etape.

Pourquoi Cloud IAM est-il essentiel pour la securite des systemes d'information ?

Cloud IAM constitue un element fondamental de la securite des systemes d'information car il permet de reduire significativement la surface d'attaque, d'ameliorer la detection des menaces et de renforcer la posture globale de securite de l'organisation face aux cybermenaces actuelles.

Quels sont les risques les plus critiques liés à Cloud IAM : Sécurisation des Identités et Accès AWS, ?

Les buckets S3 publics, les rôles IAM trop permissifs et les security groups ouverts au monde. Ces trois erreurs représentent plus de 60% des incidents cloud selon les rapports Datadog et Palo Alto.

Sources et références : CISA · Cloud Security Alliance

Articles connexes

- Container Security : Docker et Runtime Protection Avancée
- Secrets Management Cloud : Vault et Key Vault 2026

> Points clés à retenir
>
> - 8. Scénarios d'attaque IAM et remédiation
> - Questions frequentes
> - 9. Conclusion : vers une posture IAM mature

9. Conclusion : vers une posture IAM mature

La sécurité IAM dans le cloud est un **processus continu**, pas un projet ponctuel. L'identité est le plan de contrôle du cloud, et chaque permission excessive, chaque credential non roté, chaque fédération mal configurée est un vecteur d'attaque potentiel. Les organisations matures adoptent une approche holistique qui combine :

- **Moindre privilège systématique** : utiliser les outils d'analyse natifs (IAM Access Analyzer, Policy Analyzer, Entra Access Reviews) pour réduire continuellement les permissions
- **Identités fédérées et machine-native** : éliminer les credentials statiques au profit de SSO, IRSA, Managed Identities et Workload Identity
- **Accès Just-in-Time** : PIM (Azure), IAM Identity Center (AWS), PAM (GCP) pour les accès privilégiés temporaires
- **Contraintes organisationnelles** : SCP, Azure Policy, Organization Policies pour définir des garde-fous infranchissables
- **Monitoring continu** : alertes sur les événements IAM critiques via CloudTrail, Audit Logs et Entra ID Sign-in Logs
- **Tests offensifs réguliers** : simuler les scénarios d'escalade de privilèges et de mouvement latéral pour valider les contrôles

La complexité multi-cloud ajoute une dimension supplémentaire : chaque provider a ses spécificités, ses pièges et ses angles morts. La standardisation de la posture IAM à travers les clouds passe par des outils CSPM qui normalisent les contrôles et offrent une visibilité unifiée. Combinée avec une gouvernance des identités rigoureuse et des tests de sécurité réguliers, une stratégie IAM mature réduit considérablement le risque de compromission cloud.

En résumé : Dans le cloud, l'identité est tout. Une politique IAM trop permissive est l'équivalent d'un mot de passe admin sur un post-it. Appliquez le moindre privilège, éliminez les credentials statiques, monitorez chaque action IAM, et testez régulièrement votre posture -- c'est la seule approche qui résiste aux attaquants modernes.

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

ayinedjimi-consultants.fr · ayi@ayinedjimi-consultants.fr