Active Directory : configurer une stratégie d'audit avancée avec GPO

La stratégie d'audit avancée Active Directory permet de tracer précisément les authentifications, les accès aux objets sensibles et les élévations de privilèges via des GPO dédiées. Bien configuré, ce dispositif fournit les Event IDs critiques (4625, 4740, 4768, 4769, 4771, 4776) nécessaires pour détecter Kerberoasting, Pass-the-Hash, DCSync et les compromissions de comptes à privilèges.

Configurer une **stratégie d'audit avancée Active Directory** (audit avancé active directory GPO) est le fondement d'une posture défensive sérieuse. Sans audit correctement configuré, les attaques les plus sophistiquées passent inaperçues dans les journaux Windows — et les journaux Windows sont votre première source de threat intelligence interne. La plupart des incidents de sécurité AD que j'ai analysés après coup avaient laissé des traces claires dans les Event Logs : des Event ID 4769 avec chiffrement RC4 (signature du Kerberoasting), des Event ID 4624 de type 3 depuis des machines inhabituelles (Pass-the-Hash), des Event ID 4662 depuis des comptes qui ne sont pas des DC (DCSync). Le problème : ces traces existaient, mais personne ne les regardait, parce que l'audit était soit désactivé, soit tellement verbeux qu'il noyait le signal dans le bruit. L'audit de base Windows Server est insuffisant pour la détection des menaces AD modernes. Ce guide configure la stratégie d'audit avancée en 9 catégories via GPO, définit les SACL sur les OUs sensibles, configure la taille des journaux et prépare l'export vers un SIEM.

À RETENIR

À retenir

Audit avancé vs basique : les paramètres Advanced Audit Policy Configuration offrent 53 sous-catégories contre 9 catégories basiques — la granularité est essentielle pour cibler les événements utiles sans noyer les journaux.

Event ID 4625 : échec d'authentification NTLM — outil principal pour détecter brute-force, credential stuffing et password spray sur les comptes du domaine.

Event IDs 4768/4769 : demandes TGT et TGS Kerberos — indispensables pour détecter le Kerberoasting (4769 avec EncryptionType 0x17 = RC4) et l'AS-REP Roasting (4768 avec PreAuth absent).

SACL sur les OUs sensibles : les System Access Control Lists permettent d'auditer les accès LDAP aux objets AD — sans elles, les lectures d'attributs sensibles (hashes, SPN) ne sont pas tracées même avec l'audit activé.

Taille des journaux : le journal Security doit être agrandi à 1-4 Go minimum et configuré en mode "AutoBackup" avec archivage automatique pour éviter la perte d'événements critiques lors des analyses post-incident.

Audit basique vs audit avancé : la vraie différence en pratique

Windows propose deux niveaux d'audit dans les GPO. L'audit basique (*Computer Configuration > Windows Settings > Security Settings > Local Policies > Audit Policy*) a 9 catégories grossières. L'audit avancé (*Computer Configuration > Windows Settings > Security Settings > Advanced Audit Policy Configuration*) décompose chacune en **53 sous-catégories précises**, avec la possibilité d'auditer séparément succès et échecs.

Exemple concret : en audit basique, activer "Account Logon: Success" génère des milliers d'événements par jour pour des connexions légitimes, rendant la détection quasi impossible. En audit avancé, vous activez uniquement "Kerberos Service Ticket Operations: Failure" — résultat : un journal ciblé sur les demandes de tickets Kerberos échoués, beaucoup plus parlant pour la détection d'attaques.

Règle absolue : **ne jamais mélanger les deux niveaux** via des GPO différentes. Si vous définissez à la fois des paramètres d'audit basiques et avancés, les paramètres avancés écrasent les basiques, mais le comportement peut être imprévisible selon l'ordre d'application des GPO. N'utilisez que l'audit avancé sur des environnements Windows Server 2012 et supérieur.

Les 9 catégories d'audit avancé à configurer sur les DC

La configuration s'applique via une GPO liée à l'OU Domain Controllers dans GPMC :

Catégorie	Sous-catégorie	Succès	Échecs	Pourquoi
Account Logon	Credential Validation	Oui	Oui	Détection brute-force NTLM, Pass-the-Hash (Event 4776)
Account Logon	Kerberos Service Ticket Ops	Oui	Oui	Kerberoasting (4769 RC4), Silver Ticket
Account Logon	Kerberos Authentication Service	Oui	Oui	AS-REP Roasting (4768), Golden Ticket
Account Management	Security Group Management	Oui	Non	Ajout au groupe Domain Admins (Event 4728)
Account Management	User Account Management	Oui	Oui	Création/suppression comptes, reset MDP (Event 4720, 4723)
Logon/Logoff	Logon	Oui	Oui	Event 4624/4625 — authentifications réseau et locales
Policy Change	Audit Policy Change	Oui	Oui	Détecter la désactivation de l'audit (Event 4719)
Privilege Use	Sensitive Privilege Use	Oui	Oui	SeDebugPrivilege (Mimikatz), SeTcbPrivilege (Event 4672)
DS Access	Directory Service Replication	Oui	Non	Détection DCSync (Event 4662 avec Replicating Directory Changes)

Créer et déployer la GPO d'audit avancé par PowerShell

La GPO d'audit doit être créée dans GPMC et liée à l'OU Domain Controllers. Voici la configuration via PowerShell et `auditpol` :

```
 # Créer la GPO d'audit avancé
Import-Module GroupPolicy
$gpo = New-GPO -Name "AUDIT_AVANCE_DOMAINE_CONTROLLERS" `
    -Comment "Stratégie d'audit avancé — Event IDs critiques pour detection SOC"

# Lier la GPO à l'OU Domain Controllers avec priorité élevée
New-GPLink -Name "AUDIT_AVANCE_DOMAINE_CONTROLLERS" `
    -Target "OU=Domain Controllers,DC=domaine,DC=local" `
    -Order 1 -LinkEnabled Yes

# Configurer les sous-catégories d'audit via auditpol
# (A exécuter sur chaque DC ou via la GPO elle-même)

# Account Logon
auditpol /set /subcategory:"Credential Validation" /success:enable /failure:enable
auditpol /set /subcategory:"Kerberos Service Ticket Operations" /success:enable /failure:enable
auditpol /set /subcategory:"Kerberos Authentication Service" /success:enable /failure:enable

# Account Management
auditpol /set /subcategory:"Security Group Management" /success:enable /failure:disable
auditpol /set /subcategory:"User Account Management" /success:enable /failure:enable
auditpol /set /subcategory:"Computer Account Management" /success:enable /failure:disable

# Logon/Logoff
auditpol /set /subcategory:"Logon" /success:enable /failure:enable
auditpol /set /subcategory:"Logoff" /success:enable /failure:disable
auditpol /set /subcategory:"Account Lockout" /success:enable /failure:enable

# Privilege Use
auditpol /set /subcategory:"Sensitive Privilege Use" /success:enable /failure:enable

# Policy Change
auditpol /set /subcategory:"Audit Policy Change" /success:enable /failure:enable
auditpol /set /subcategory:"Authentication Policy Change" /success:enable /failure:disable

# DS Access (activer seulement avec SACL configurées)
auditpol /set /subcategory:"Directory Service Replication" /success:enable /failure:disable
auditpol /set /subcategory:"Directory Service Access" /success:enable /failure:enable

# Vérifier la configuration
auditpol /get /category:* | Where-Object { $_ -match "(Success|Failure)" }
```

Quels sont les Event IDs critiques à surveiller en priorité ?

Ces Event IDs sont générés dans le journal Security des DC. Ils constituent la base des alertes SOC pour la détection des attaques AD :

Event 4625 — Échec de connexion. Le champ "Failure Reason" distingue mauvais MDP, compte verrouillé, compte inconnu. Un pic sur un même compte = brute-force. Plusieurs comptes différents en courte période = password spray ou credential stuffing.

Event 4740 — Compte verrouillé. Le champ "Caller Computer Name" identifie la machine source — outil pour le helpdesk ET pour détecter les outils automatisés de bruteforce qui verrouillent des dizaines de comptes.

Event 4768 — Demande de TGT Kerberos (AS-REQ). Préauthentification désactivée sur un compte + cet événement = AS-REP Roasting. Filtrer sur EncryptionType = 0x17 (RC4) ou absence du champ PreAuth.

Event 4769 — Demande de ticket TGS Kerberos. EncryptionType = 0x17 (RC4) pour un compte de service avec SPN = Kerberoasting en cours. C'est l'Event ID de référence pour détecter cette attaque.

Event 4771 — Échec de pré-authentification Kerberos. Failure Code 0x18 = mauvais mot de passe, 0x25 = horloge désynchronisée (signe d'une VM clonée ou de manipulation du temps).

Event 4776 — Validation de credentials NTLM. Succès et échecs des authentifications NTLM. Authentification NTLM depuis une workstation vers un serveur sensible hors des heures ouvrées = signal Pass-the-Hash potentiel.

Configurer les SACL sur les OUs et objets sensibles

Les *SACL (System Access Control Lists)* définissent quels accès aux objets AD doivent être audités. Sans SACL, même avec l'audit "Directory Service Access" activé dans la GPO, aucun accès LDAP aux objets de l'annuaire n'est tracé. La GPO active la catégorie d'audit, les SACL définissent sur quels objets.

Priorités d'implémentation :

Groupe Domain Admins et Enterprise Admins — auditer tout accès en écriture (ajout/suppression de membres)

Objet AdminSDHolder — auditer les modifications de permissions qui affectent tous les objets protégés

Comptes Tier 0 (krbtgt, DC machine accounts) — auditer tout accès en lecture des attributs sensibles

Objet domaine racine — auditer les accès aux droits de réplication (détection DCSync)

```
 # Configurer une SACL sur le groupe Domain Admins
# Auditer toutes les modifications de membres (ajouts/suppressions)
$domainDN = (Get-ADDomain).DistinguishedName
$adminsDN = "CN=Domain Admins,CN=Users,$domainDN"

$acl = Get-Acl "AD:$adminsDN"

# Créer une règle d'audit SACL — Everyone, WriteProperty (modifications), Success et Failure
$identity = [System.Security.Principal.NTAccount]"Everyone"
$rights = [System.DirectoryServices.ActiveDirectoryRights]::WriteProperty
$type = [System.Security.AccessControl.AuditFlags]::Success -bor `
        [System.Security.AccessControl.AuditFlags]::Failure
$inheritanceType = [System.DirectoryServices.ActiveDirectorySecurityInheritance]::None

$auditRule = New-Object System.DirectoryServices.ActiveDirectoryAuditRule(
    $identity, $rights, $type, $inheritanceType
)
$acl.AddAuditRule($auditRule)
Set-Acl "AD:$adminsDN" $acl
Write-Host "SACL configurée sur $adminsDN — modifications de membres auditées"

# Configurer la SACL sur l'objet domaine pour détecter DCSync
# Auditer l'accès au droit de contrôle "Replicating Directory Changes"
$domainObj = "AD:$domainDN"
$acl = Get-Acl $domainObj
# Le GUID de "Replicating Directory Changes" est 1131f6aa-9c07-11d1-f79f-00c04fc2dcd2
$replicationGuid = [System.Guid]"1131f6aa-9c07-11d1-f79f-00c04fc2dcd2"
$rights = [System.DirectoryServices.ActiveDirectoryRights]::ExtendedRight
$type = [System.Security.AccessControl.AuditFlags]::Success
$auditRule = New-Object System.DirectoryServices.ActiveDirectoryAuditRule(
    $identity, $rights, $type, $replicationGuid, "None"
)
$acl.AddAuditRule($auditRule)
Set-Acl $domainObj $acl
Write-Host "SACL DCSync configurée sur $domainDN"
```

Configurer la taille et la rétention des journaux Security

Un audit bien configuré produit des événements — encore faut-il qu'ils ne soient pas écrasés avant d'être analysés. Par défaut, le journal Security Windows est limité à **20 Mo**, soit quelques heures d'événements sur un DC actif. C'est très insuffisant pour toute investigation post-incident sérieuse.

```
 # Configurer la taille maximale du journal Security à 2 Go
# et l'archivage automatique à la rotation
wevtutil sl Security /ms:2147483648 /rt:true /ab:true
# /ms = max size en octets (2 Go = 2 * 1024^3 = 2 147 483 648)
# /rt = retain (true = archiver avant d'écraser)
# /ab = auto-backup vers le chemin d'archivage

# Via PowerShell (Windows Server 2016+)
$log = Get-WinEvent -ListLog Security
$log.MaximumSizeInBytes = 2GB
$log.LogMode = [System.Diagnostics.Eventing.Reader.EventLogMode]::AutoBackup
$log.SaveChanges()

# Déployer ce paramètre via GPO sur tous les DC
# Computer Configuration > Windows Settings > Security Settings > Event Log
# Maximum security log size = 2097152 KB (2 Go)
# Retention method for security log = Overwrite events as needed (archive first)

# Vérifier la configuration actuelle
wevtutil gl Security | Select-String "maxSize|logMode"
```

Exporter les journaux vers un SIEM pour la corrélation

Les journaux en silo sur chaque DC n'ont qu'une valeur limitée. La corrélation entre plusieurs DC, avec des données réseau et endpoint, nécessite une centralisation dans un SIEM. Les solutions principales :

Microsoft Sentinel : connecteur natif via Azure Monitor Agent (AMA) — ingestion directe des Event IDs sélectionnés. KQL pour les requêtes de détection. Option recommandée pour les organisations déjà dans l'écosystème Azure.

Wazuh : agent open-source léger, décodeur XML intégré pour les Event IDs Windows, règles de détection prédéfinies pour l'AD.

Windows Event Forwarding (WEF) : solution native Windows, zéro agent tiers, centralise vers un Windows Event Collector (WEC) via les GPO standards — sous-estimée et très efficace pour les environnements qui ne veulent pas de dépendance cloud.

Pour détecter le Kerberoasting avec Microsoft Sentinel, la requête KQL de base est : `SecurityEvent | where EventID == 4769 | where TicketEncryptionType == "0x17"`. Notre article sur le threat hunting avec Microsoft Sentinel détaille la configuration complète du connecteur Active Directory et les règles de détection pour les attaques AD les plus communes. Pour un audit de posture AD complet, voir notre article sur les outils d'audit Active Directory.

Détecter la désactivation de l'audit par un attaquant

Un attaquant expérimenté cherche à désactiver l'audit dès qu'il obtient des droits suffisants sur les DC — pour effacer ses traces ou éviter la détection pendant la phase d'exfiltration. C'est pourquoi l'audit des changements de politique d'audit (Event ID **4719**) est critique et doit être configuré en tout premier.

Si quelqu'un modifie la politique d'audit — pour désactiver l'Event 4769 avant de faire du Kerberoasting, par exemple — l'Event 4719 est généré avant la désactivation. L'alerte SIEM

prioritaire : Event 4719 depuis un compte non-DC, non-service connu → incident potentiel à investiguer immédiatement.

```
 # Chercher les tentatives de désactivation d'audit dans les 72 dernières heures
$since = (Get-Date).AddHours(-72)
Get-WinEvent -FilterHashtable @{
    LogName = 'Security'; Id = 4719; StartTime = $since
} | Select-Object TimeCreated,
    @{N='Utilisateur';E={$_.Properties[1].Value}},
    @{N='Machine';E={$_.Properties[3].Value}},
    @{N='Changement';E={$_.Properties[4].Value}} | Format-Table -AutoSize

# Chercher les tentatives de Kerberoasting (TGS avec chiffrement RC4)
Get-WinEvent -FilterHashtable @{
    LogName = 'Security'; Id = 4769; StartTime = $since
} | Where-Object {
    # TicketEncryptionType = 0x17 = RC4 (signature Kerberoasting)
    $_.Properties[5].Value -eq "0x17"
} | Select-Object TimeCreated,
    @{N='Compte cible SPN';E={$_.Properties[0].Value}},
    @{N='Demandeur';E={$_.Properties[1].Value}},
    @{N='Machine source';E={$_.Properties[7].Value}} | Format-Table -AutoSize

# Chercher les verrouillages de comptes (password spray)
Get-WinEvent -FilterHashtable @{
    LogName = 'Security'; Id = 4740; StartTime = $since
} | Group-Object { $_.Properties[0].Value } |
    Sort-Object Count -Descending |
    Select-Object Name, Count | Format-Table -AutoSize
```

Pour aller plus loin sur la détection des attaques AD, notre article sur les 10 attaques Active Directory détaille les patterns d'attaque et leurs Event IDs correspondants. Les Event IDs AD sont également couverts dans notre guide de sécurité Active Directory. Dans le cadre de la certification ISO 27001:2022 (contrôle A.8.15 — journalisation) et de la conformité NIS 2 (article 21 — mesures de surveillance), la configuration d'un audit AD avancé avec export SIEM est une exigence documentaire et technique centrale que notre service RSSI externalisé met en place systématiquement.

Implémenter Windows Event Forwarding (WEF) sans SIEM tiers

La solution WEF native Windows est souvent sous-estimée. Elle ne nécessite ni agent supplémentaire ni licence SIEM — juste les GPO Windows et un serveur Windows Server dédié en tant que **Windows Event Collector (WEC)**. Pour les PME qui n'ont pas les moyens d'un Splunk ou d'un Sentinel, c'est une option sérieuse et production-ready.

L'architecture est simple : chaque DC est configuré comme "Source" qui envoie ses événements vers le WEC via le protocole WS-Management (port 5985 HTTP ou 5986 HTTPS). Le WEC collecte les événements dans ses journaux locaux, lisibles avec Event Viewer ou via PowerShell.

```
 # Configuration WEF côté serveur WEC (Windows Event Collector)
# Activer le service Windows Event Collector
Start-Service Wecsvc
Set-Service Wecsvc -StartupType Automatic

# Configurer le WEC comme collecteur
wecutil qc /q

# Créer un abonnement WEF pour les Event IDs critiques AD
# Le fichier XML définit les sources et les filtres d'événements
$subscriptionXml = @"

  CriticalAD-Events
  SourceInitiated
  Event IDs critiques Active Directory pour detection SOC
  true
  http://schemas.microsoft.com/wbem/wsman/1/windows/EventLog
  MinLatency

    30000




        *[System[(EventID=4625 or
        EventID=4740 or EventID=4768 or
        EventID=4769 or EventID=4771 or
        EventID=4776 or EventID=4719 or
        EventID=4728 or EventID=4732 or
        EventID=4720 or EventID=4672 or
        EventID=4662)]]


  ]]>
  false
  http

  ForwardedEvents

  O:NSG:NSD:P(A;;GA;;;DC)S:

"@
```

```
$subscriptionXml | Out-File -FilePath "C:\Temp\CriticalAD.xml" -Encoding UTF8
wecutil cs "C:\Temp\CriticalAD.xml"
Write-Host "Abonnement WEF créé — collecte des Event IDs AD critiques"

# Configuration GPO côté DC source (à appliquer sur l'OU Domain Controllers)
# Computer Configuration > Windows Settings > Security Settings > System Services
# Windows Remote Management (WinRM) = Automatic + Allow
# Sous Computer Configuration > Administrative Templates > Windows Components > Event Forwarding
# Configure the server address = http://WEC-SERVER:5985/wsman/SubscriptionManager/WEC
```

Créer des rapports d'audit périodiques pour la direction et les auditeurs

La configuration de l'audit AD ne sert pas uniquement à la détection temps réel — elle sert aussi à produire des preuves documentaires pour les audits de conformité ISO 27001, NIS 2, RGPD. Les auditeurs demandent systématiquement des rapports sur l'accès aux comptes à privilèges, les réinitialisations de mots de passe et les modifications de groupes sensibles.

Un script PowerShell de reporting mensuel qui extrait les événements clés et les exporte en Excel ou HTML facilite considérablement ces audits. Il documente proactivement que le contrôle de surveillance est en place et fonctionnel, transformant l'audit en formalité plutôt qu'en crise.

```
 # Rapport mensuel d'audit AD — export CSV pour auditeurs
param(
    [DateTime]$StartDate = (Get-Date).AddDays(-30),
    [DateTime]$EndDate   = Get-Date,
    [string]$ReportPath  = "C:\Reports\AD-Audit-$(Get-Date -Format 'yyyy-MM').csv"
)

$results = [System.Collections.ArrayList]::new()

# Nouveaux comptes créés
$newAccounts = Get-WinEvent -FilterHashtable @{
    LogName='Security'; Id=4720; StartTime=$StartDate; EndTime=$EndDate
} -ErrorAction SilentlyContinue | ForEach-Object {
    [PSCustomObject]@{
        Type = "Compte créé"
        Date = $_.TimeCreated
        Sujet = $_.Properties[0].Value
        ParCompte = $_.Properties[4].Value
        Machine = $_.Properties[6].Value
    }
}
if ($newAccounts) { [void]$results.AddRange($newAccounts) }

# Modifications de groupes à privilèges (Domain Admins = 512)
$groupChanges = Get-WinEvent -FilterHashtable @{
    LogName='Security'; Id=4728; StartTime=$StartDate; EndTime=$EndDate
} -ErrorAction SilentlyContinue | ForEach-Object {
    [PSCustomObject]@{
        Type = "Ajout groupe privilégié"
        Date = $_.TimeCreated
        Sujet = $_.Properties[0].Value
        ParCompte = $_.Properties[4].Value
        Machine = $_.Properties[6].Value
    }
}
if ($groupChanges) { [void]$results.AddRange($groupChanges) }

# Réinitialisations de mots de passe (Event 4723 = utilisateur change son MDP)
$pwdResets = Get-WinEvent -FilterHashtable @{
    LogName='Security'; Id=4723; StartTime=$StartDate; EndTime=$EndDate
} -ErrorAction SilentlyContinue | Select-Object -First 200 | ForEach-Object {
    [PSCustomObject]@{
        Type = "Réinitialisation MDP"
```

```
        Date = $_.TimeCreated
        Sujet = $_.Properties[0].Value
        ParCompte = $_.Properties[4].Value
        Machine = $_.Properties[6].Value
    }
}
if ($pwdResets) { [void]$results.AddRange($pwdResets) }

# Exporter le rapport
$results | Sort-Object Date | Export-Csv -Path $ReportPath -Encoding UTF8 -NoTypeInformation
Write-Host "Rapport d'audit exporté : $($results.Count) événements dans $ReportPath"
```

Ce rapport, archivé mensuellement, constitue une preuve documentaire solide pour les auditeurs ISO 27001 et pour les déclarations d'incidents NIS 2. Il complète notre guide sur les GPO de sécurisation Active Directory et s'inscrit dans la stratégie globale de supervision documentée dans notre hub Active Directory.

Questions fréquentes sur l'audit avancé Active Directory

Quelle est la différence entre l'audit avancé via GPO et auditpol.exe ?

Les deux agissent sur la même couche d'audit Windows via des mécanismes différents. **auditpol.exe** modifie directement la politique locale — ses paramètres peuvent être écrasés par les GPO. La **GPO Advanced Audit Policy** s'applique via le mécanisme Group Policy et prévaut sur les paramètres locaux. Pour un environnement AD, toujours préférer les GPO. auditpol est utile pour des tests rapides en local ou sur des machines standalone non membres d'un domaine.

L'activation de tous ces audits va-t-elle surcharger les DC ?

L'impact CPU est généralement faible — inférieur à 1-2% sur un DC moderne. L'impact principal porte sur l'I/O disque et la taille des journaux. Sur les DC gérant plus de 5000 authentifications par heure, une politique trop large (succès ET échecs sur tout) peut générer plusieurs gigaoctets de logs par jour. La règle : auditer les *échecs* plus largement que les *succès*,

et n'activer "Directory Service Access Success" que sur des objets ciblés via SACL. Avec la bonne granularité, l'impact est maîtrisable même sur des DC chargés.

Comment identifier les comptes vulnérables au Kerberoasting via les Event IDs ?

L'Event ID 4769 avec `Ticket Encryption Type = 0x17` (RC4-HMAC) signale une demande de ticket TGS avec RC4. Les comptes de service ayant un SPN et un mot de passe faible sont les cibles. Filtrez les 4769 RC4 provenant de machines qui ne sont pas des DC — une workstation demandant des TGS RC4 hors des heures ouvrées est un signal fort de Kerberoasting actif. En pratique, un attaquant génère souvent plusieurs dizaines de 4769 en quelques secondes pour tous les comptes avec SPN du domaine.

Faut-il des droits Domain Admin pour configurer les SACL sur les OUs ?

Pour les OUs et objets standards, il faut les droits *Modify Permissions* sur l'objet cible. Pour les objets protégés par AdminSDHolder (Domain Admins, krbtgt, DC accounts), seuls les **Domain Admins ou Enterprise Admins** peuvent modifier les SACL. Pour l'objet domaine racine, seul Enterprise Admins peut modifier les SACL. La délégation fine est possible mais complexe à maintenir — à documenter dans votre plan de gestion des privilèges Tier 0.

Comment s'assurer que l'audit ne peut pas être contourné par un attaquant ?

Quatre mesures complémentaires : (1) Restreindre l'accès local aux DC aux comptes Tier 0 uniquement — un attaquant doit d'abord compromettre un compte admin DC pour modifier l'audit. (2) Monitorer Event 4719 en temps réel avec alerte SIEM immédiate. (3) Configurer via GPO le droit "Manage auditing and security log" en restreignant à Administrators uniquement. (4) Envoyer les logs vers un SIEM externe en temps réel — même si un attaquant efface les journaux locaux, les copies SIEM restent intactes. Cette dernière mesure est la plus importante : un SIEM bien configuré invalide totalement l'intérêt de la suppression de logs locaux pour un attaquant.