

ITSec

SOMMET DE
LA SÉCURITÉ
INFORMATIQUE

PRÉSENTÉ PAR DEVOLUTIONS ET SHERWEB

JOURNÉE DE
FORMATION

**Démystifier
l'authentification RDP :
techniques avancées de
diagnostic**

Téléchargez votre application *Whova*

Connectez-vous à votre profil en entrant l'adresse courriel utilisée lors de votre inscription.

Whova vous permet de :

Vous mettre présent dans la formation suivie

Recevoir votre **badge de certification!**



Télécharger dans
l'App Store



DISPONIBLE SUR
Google Play



Introduction

Marc-André Moreau
Directeur de la technologie chez Devolutions

Expert du protocole RDP
Contributeur au logiciel libre
Adepté du *reverse engineering*



Survol des notions

- Journaux d'événements
- Niveaux de sécurité RDP
- Authentification du serveur RDP
- *Network Level Authentication*
- Couche de transport RDP UDP
- Configuration TLS dans RDP
- RDP NLA sans délégation
- RDP Azure AD / Entra ID
- RDP NLA avec Kerberos
- Configuration DNS
- Détection du KDC
- Kerberos sur macOS
- Carte à puce virtuelle Windows
- Proxy KDC de Kerberos
- Remote Desktop Gateway

Environnement de laboratoire

Laboratoire Active Directory avec Hyper-V automatisé avec PowerShell :

<https://github.com/Devolutions/devolutions-labs>

IT-HELP-RTR : Alpine Linux servant de routeur avec DHCP

IT-HELP-DC : Windows Server 2022 + contrôleur de domaine + AD CS + DNS

IT-HELP-TEST : Windows Server 2022, serveur RDP de test joint au domaine

IT-HELP-CLIENT : Windows 11 23H2 non joint au domaine, mais sur le même réseau

Suffixe DNS : ad.it-help.ninja

Kerberos realm : ad.it-help.ninja

Nom de domaine : ad.it-help.ninja (IT-HELP)

Comptes de test :

[Administrator@ad.it-help.ninja](#) – administrateur du domaine

[ProtectedUser@ad.it-help.ninja](#) – membre du groupe *Protected Users*

Règles de survie RDP

1. Les messages d'erreur sont souvent les mêmes peu importe la cause
 - Il faut donc sortir une boule de cristal et user de beaucoup d'imagination
2. Ne cherchez pas les outils de diagnostic officiels, ils n'existent pas
 - J'ai menti – Microsoft a ses propres outils internes qu'ils gardent pour eux
3. Les outils Sysinternals, NirSoft, IDA et Wireshark valent leur pesant d'or
 - Un peu de *reverse engineering* vous épargne des heures de recherche infructueuse
 - Oui, je dis ça le plus sérieusement du monde : la documentation aide si peu
4. En cas de doute, sortir l'artillerie lourde même si c'est plus de travail à utiliser
 - Déchiffrement TLS dans Wireshark (<https://github.com/awakecoding/wireshark-rdp>)
 - Journaux internes du client par *API hooking* (<https://github.com/Devolutions/MsRdpEx>)

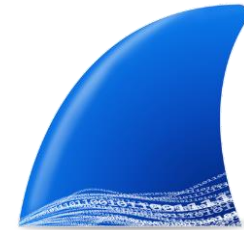
Wireshark avec RDP

Téléchargez le contenu du répertoire git : <https://github.com/awakecoding/wireshark-rdp>

- Installez et lancez Wireshark (<https://www.wireshark.org/>)
- Ouvrez les fichiers captures RDP déchiffrées de référence (dossier « captures »)
- Optionnel : suivez les instructions pour déchiffrer votre propre trafic RDP

Même sans déchiffrement de trafic, Wireshark est utile pour les diagnostics :

- On peut voir l'ouverture et surtout la fermeture de connexion TCP.
 - Gardez l'œil ouvert pour le paquet TCP RST, TCP FIN
- On peut voir l'ouverture et la fermeture TLS, le nombre messages envoyés, etc.
 - Gardez l'œil ouvert pour le message TLS Alert, même si le contenu est chiffré
- On peut souvent identifier qui, entre le client et le serveur, initie la déconnexion.



Gardez le dossier de captures Wireshark à portée de main pour mieux suivre le contenu de la formation, en consultant les exemples correspondants aux différents scénarios de connexion.

MsRdpEx : client RDP de Microsoft amélioré



Téléchargez et installez MsRdpEx : <https://github.com/Devolutions/MsRdpEx>

- Suivez les instructions pour l'utilisation des options de fichier .RDP étendues.
- Lancez MsRdpEx avec les variables d'environnement pour la journalisation.

MsRdpEx lance mstsc (ou msrdc) en injectant une DLL avec *Detours* [pour faire du API hooking](#).

- En gros, on modifie le comportement interne pour ajouter des fonctionnalités manquantes
- On ajoute aussi de la journalisation du comportement et variables internes du client RDP!

Journaux d'évènements

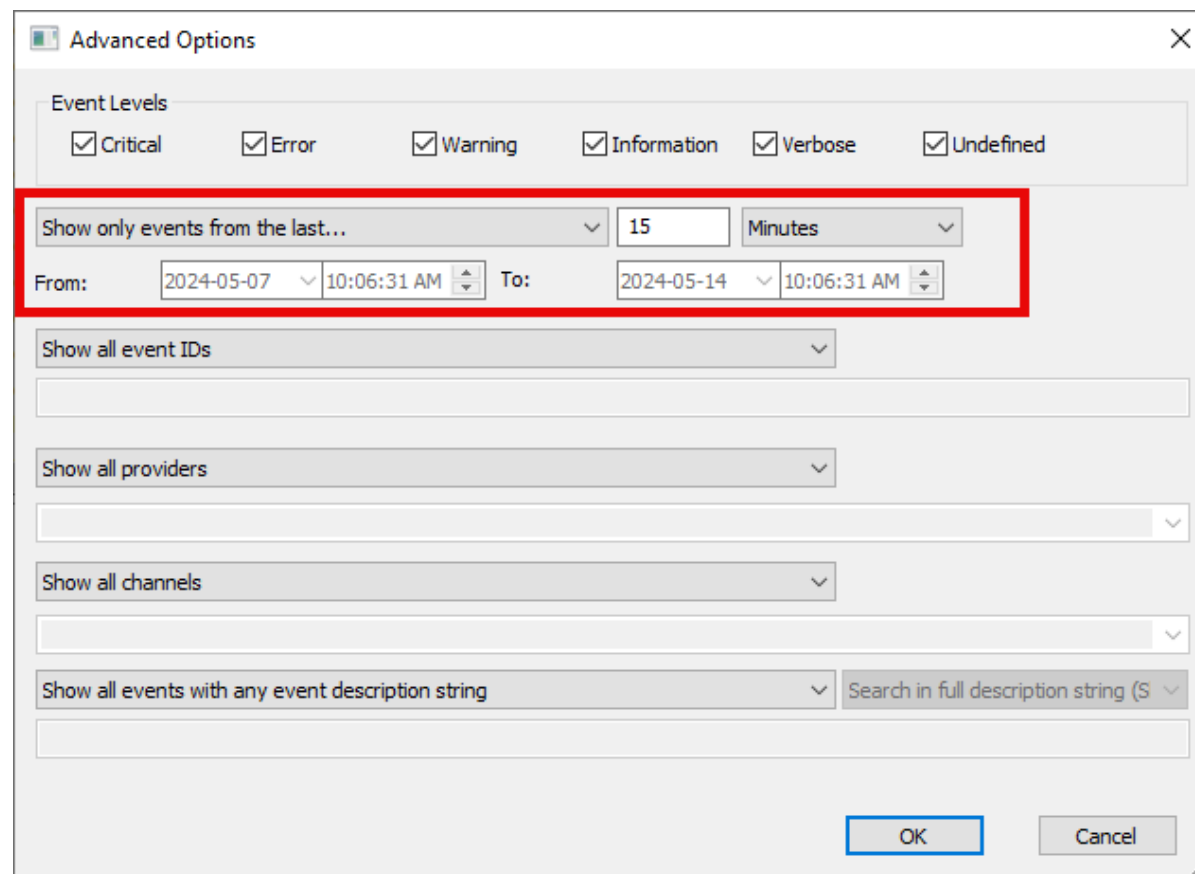
- Outil Nirsoft FullEventLogView
- Évènements côté client RDP
- Évènements côté serveur RDP
- Autres évènements Remote Desktop Services

Outil Nirsoft FullEventLogView

Windows Event Viewer est lent et pénible à utiliser, sans option de recherche globale.

L'outil [Nirsoft FullEventLogView](#) vous aide à mieux chercher une aiguille dans une botte de foin.

Lors du premier lancement, allez dans **Options**, puis **Advanced Options** pour charger seulement les dernières 15 minutes d'évènements au lieu des derniers 7 jours, ce qui sera beaucoup plus rapide.



Évènements côté client

Dans le *Windows Event Viewer*, dans la section **Applications and Services Logs\Microsoft\Windows:**

- **TerminalServices-ClientActiveXCore**
 - Erreurs de déconnexion, problèmes d'engin de décodage graphique.
- **Security-Kerberos**
 - Désactivé par défaut, donne des indices pour les problèmes avec Kerberos.
- **CAPI2**
 - Le détail des problèmes de validation de certificats se retrouve ici.

La plupart des problèmes avec le client RDP n'ont pas d'évènements détaillés dans le *Windows Event Viewer*.

Évènements côté serveur

Dans le *Windows Event Viewer*, dans la section **Applications and Services Logs\Microsoft\Windows :**

- **TerminalServices-RemoteConnectionManager**
 - Évènements en lien avec les sessions Windows à distance (donc RDP)
- **TerminalServices-LocalSessionManager**
 - Évènements en lien avec les sessions Windows « locales », incluant RDP.
- **RemoteDesktopServices-RdpCoreTS**
 - Évènements en lien avec les canaux virtuels, les fermetures de session, etc.

Il existe beaucoup d'autres journaux d'évènements côté serveur, mais il est difficile de trouver les bons.

Il ne faut pas oublier d'activer ceux qui ne sont pas activés par défaut.

Autres évènements Remote Desktop Services

Dans le *Windows Event Viewer*, dans la section **Applications and Services Logs\Microsoft\Windows** :

- **TerminalServices-Gateway**
 - Évènements d'ouverture et fermeture de tunnels avec RD Gateway
- **TerminalServices-Licensing**
 - Évènements en lien avec la gestion de licences (RDS CALs)
- **TerminalServices-SessionBroker**
 - Évènements en lien avec le *session brokering* (RDS Farm)

Encore une fois, il existe beaucoup d'autres journaux d'évènements, mais ceux-ci sont les plus pertinents.

Niveaux de sécurité RDP

- Survol des niveaux de sécurité RDP
- Négociation du niveau de sécurité RDP
- Identification du niveau de sécurité RDP

Niveaux de sécurité RDP

- Archaïque (« Standard »)
 - Couche cryptographique utilisant RC4 et une clé « privée » documentée publiquement, vulnérable aux attaques MiTM. **À éviter à tout prix.**
- Transport Layer Security (TLS)
 - Couche cryptographique TLS standard, mais avec authentification Winlogon interactive. **Vulnérable aux attaques par déni de service.**
- Network Level Authentication (NLA)
 - Couche cryptographique TLS standard, avec authentification faite au niveau réseau par CredSSP avant la création de la session interactive.

Négociation du niveau de sécurité RDP

- Paquets envoyés sur TCP/3389 **avant** que TLS débute
- X.224 Connection Request, champ **requestedProtocols**
 - [\[MS-RDPBCGR\] RDP Negotiation Request \(RDP_NEG_REQ\)](#)
 - Contient la liste des niveaux de sécurité pris en charge par le client
- X.224 Connection Confirm, champ **selectedProtocol**
 - [\[MS-RDPBCGR\] RDP Negotiation Response \(RDP_NEG_RSP\)](#)
 - Contient le niveau de sécurité final sélectionné par le serveur

Wireshark est une bonne façon d'identifier le niveau de sécurité RDP négocié.

- Aucun déchiffrement TLS nécessaire puisque c'est envoyé en clair sur le réseau.

X.224 Connection Request

Wireshark · Packet 6 · vEthernet (LAN Switch)

- > Transmission Control Protocol, Src Port: 50176, Dst Port: 3389, Seq: 1, Ack: 1, Len: 19
- > TPKT, Version: 3, Length: 19
- > ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
- ▼ Remote Desktop Protocol
 - Type: RDP Negotiation Request (0x01)
 - > Flags: 0x00
 - Length: 8
 - ▼ requestedProtocols: 0x0000000b, TLS security supported, CredSSP supported, CredSSP with Early User Authorization Result PDU supported
 -1 = TLS security supported: True
 - ...1. = CredSSP supported: True
 - ...0.. = RDSTLS supported: False
 - ...1... = CredSSP with Early User Authorization Result PDU supported: True

0000	00 15 5d 07 7d 08 00 15 5d 07 7d 00 08 00 45 00	..].}...].}...E.
0010	00 3b 92 31 40 00 80 06 00 00 0a 0a 00 01 0a 0a	.;1@... ..
0020	00 08 c4 00 0d 3d cd 49 70 96 14 92 a0 eb 50 18=I p...P.
0030	04 02 14 4a 00 00 03 00 00 13 0e e0 00 00 00 00	...J... ..
0040	00 01 00 08 00 0b 00 00 00

requestedProtocols (rdp.negReq.requestedProtocols), 4 bytes

☒ Show packet bytes

Close Help

X.224 Connection Confirm

Wireshark · Packet 8 · vEthernet (LAN Switch)

> TPKT, Version: 3, Length: 19
> ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
✓ Remote Desktop Protocol
 Type: RDP Negotiation Response (0x02)
 ✓ Flags: 0x1f, Extended Client Data Blocks supported, Graphics Pipeline Extension Protocol supported, Restricted admin mode supported, Res:
 1 = Extended Client Data Blocks supported: True
 1. = Graphics Pipeline Extension Protocol supported: True
 1... = Restricted admin mode supported: True
 1 = Restricted authentication mode supported: True
 Length: 8
 selectedProtocol: CredSSP with Early User Authorization Result PDU (0x00000008)

0010	00 3b a8 43 40 00 80 06	3e 5d 0a 0a 00 08 0a 0a	;·C@· · · >] ·····
0020	00 01 0d 3d c4 00 14 92	a0 eb cd 49 70 a9 50 18	···=···· ···Ip·P·
0030	f9 ed 99 d7 00 00 03 00	00 13 0e d0 00 00 12 34	·········· 4
0040	00 02 1f 08 00 08 00 00	00	······ ·

selectedProtocol (rdp.negReq.selectedProtocol), 4 bytes

☒ Show packet bytes

Close Help

Authentification du serveur RDP

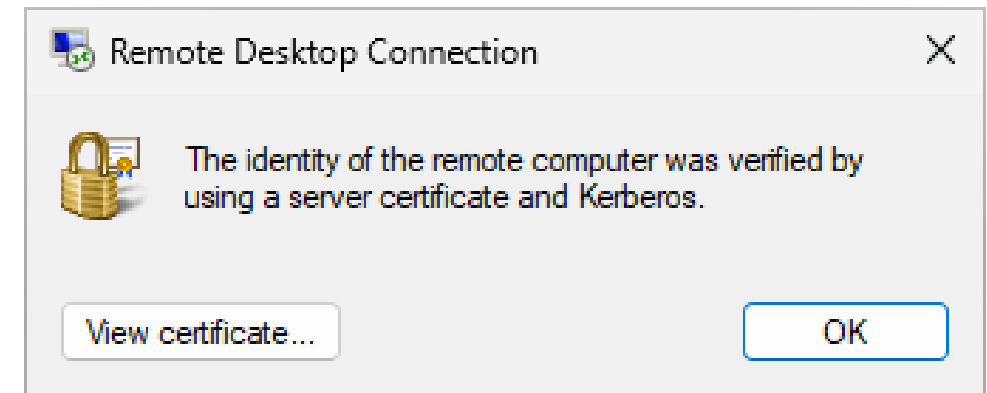
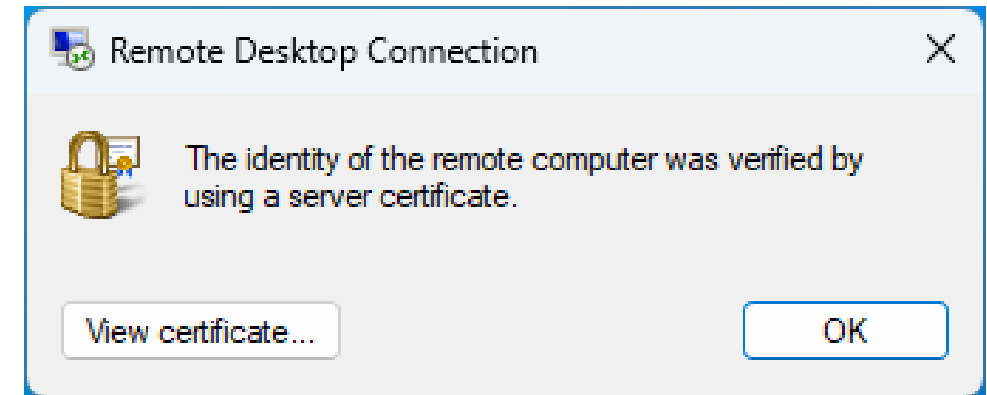
- Types d'authentification du serveur RDP
- Niveau d'authentification du serveur RDP

Types d'authentification du serveur RDP

L'authentification du **serveur** RDP sert à s'assurer que la connexion s'établisse avec le serveur de destination attendu.

Le type d'authentification serveur RDP peut s'effectuer à l'aide du certificat TLS du serveur ou par Kerberos (SPN TERMSRV/), ou par une combinaison des deux méthodes :

- 0 : Aucune authentification serveur
- 1 : Authentification par certificat
- 2 : Authentification par Kerberos
- 3 : Authentification par certificat et Kerberos



Niveau d'authentification du serveur RDP

L'option **Authentication level** du fichier .RDP contrôle le comportement du client lors d'un échec d'authentification du serveur.

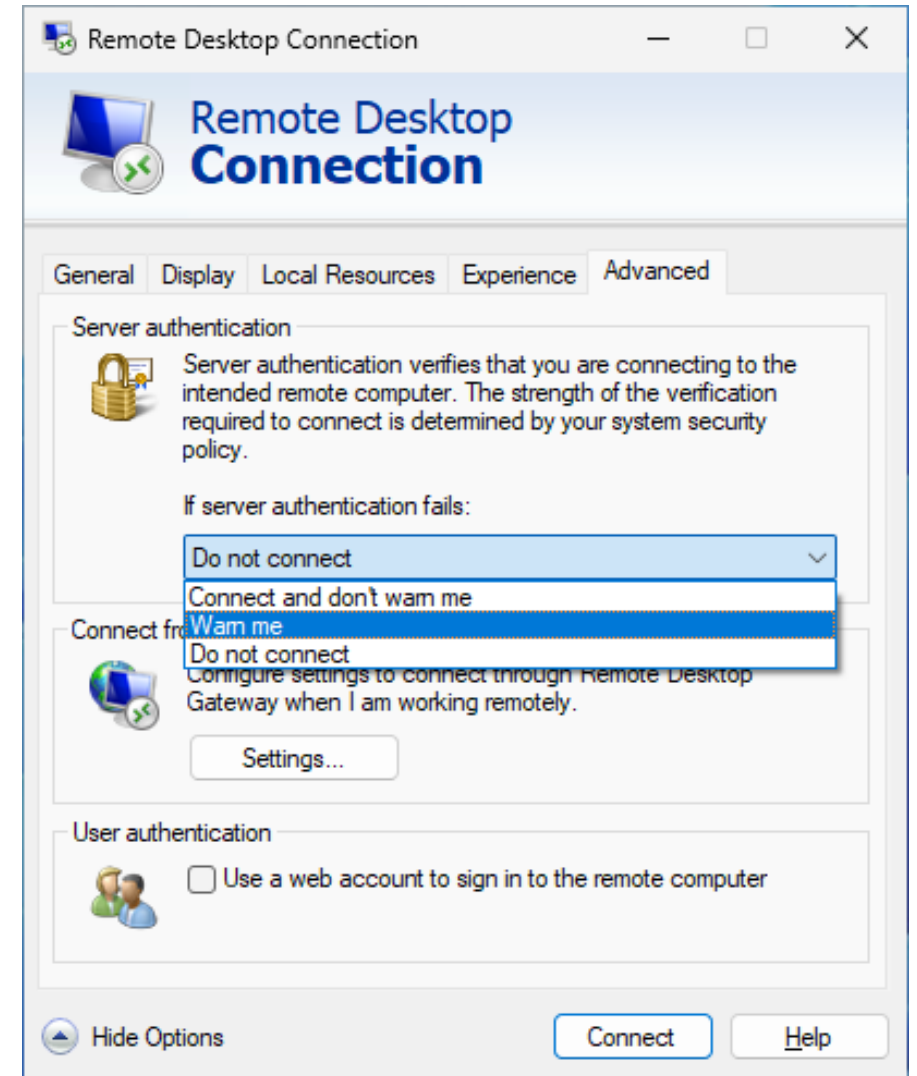
0 : Connect and don't warn me

1 : Do not connect

2 : Warn me

Veuillez noter que le **niveau** d'authentification du serveur ne fait pas la distinction du **type** d'authentification serveur utilisé.

Par certificat ou Kerberos, un seul des deux types suffit.

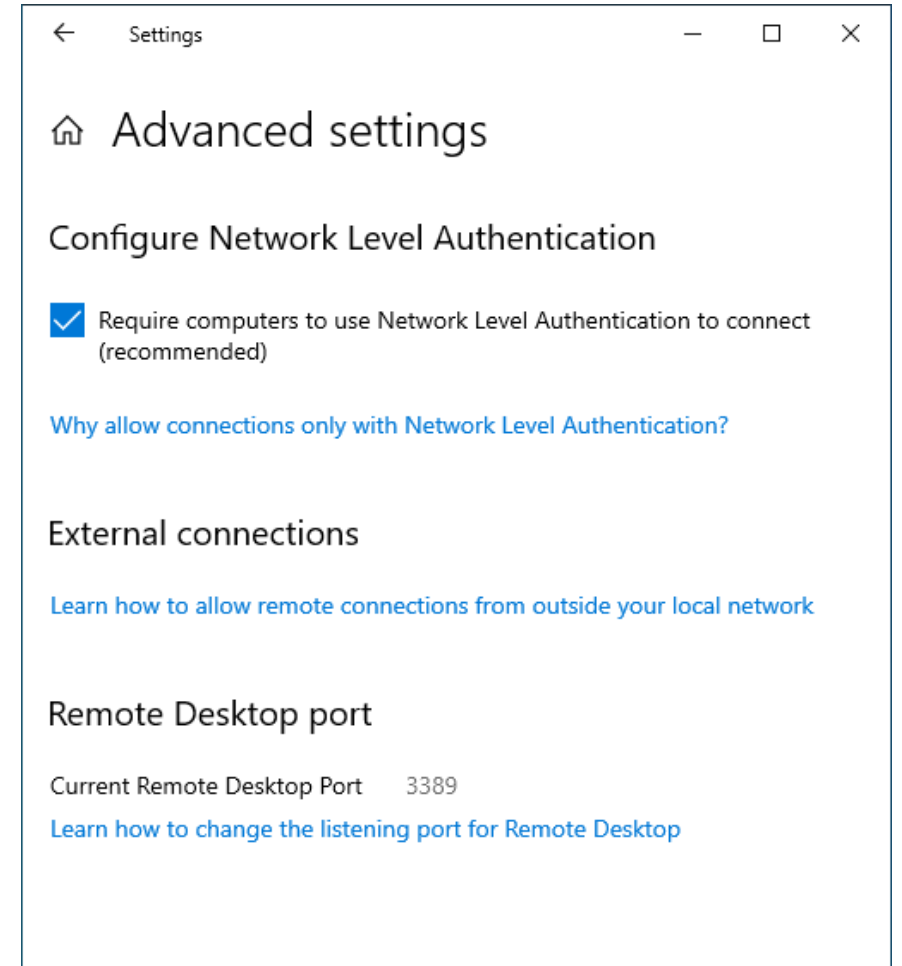


Network Level Authentication

- Origines du *Network Level Authentication* (NLA)
- Survol du protocole CredSSP utilisé dans NLA
- Désactivation RDP NLA côté client et serveur

Network Level Authentication (NLA)

- Authentification au niveau réseau en **début** de connexion
 - L'allocation des ressources pour la session interactive se fait **après** l'authentification
 - Empêche les attaques par déni de service
- Paramètre « Require computers to use Network Level Authentication to connect »
- Activé par défaut depuis Windows Vista
- **Recommandé en tout temps pour des raisons de sécurité**

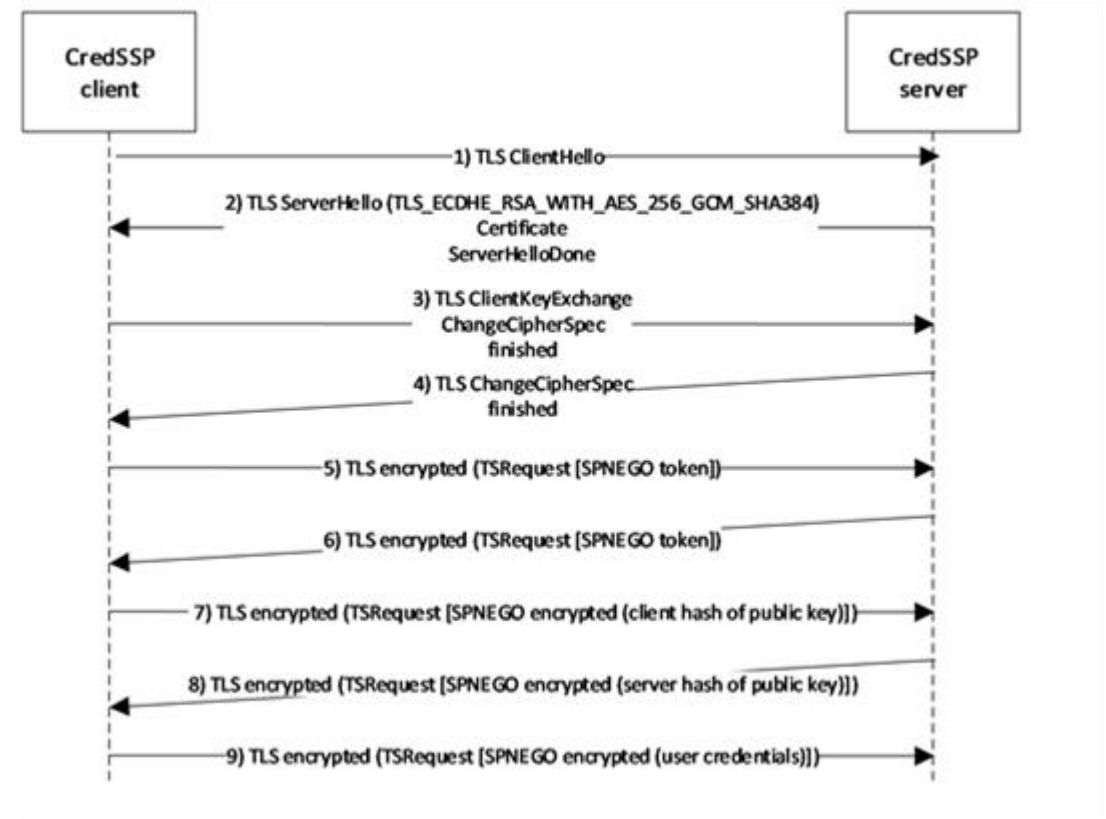


NLA : Séquence CredSSP

1. Ouverture du canal sécurisé TLS
2. Négociation NTLM ou Kerberos
3. Échange NTLM ou Kerberos
4. Répétition clé publique du serveur
5. Délégation des identifiants complets

Points importants :

- Windows Authentication (NTLM/Kerberos)
- Lien fort entre authentification et couche TLS
- Mécanisme anti-MiTM intégré (*public key echo*)
- Identifiants *complets* envoyés au serveur



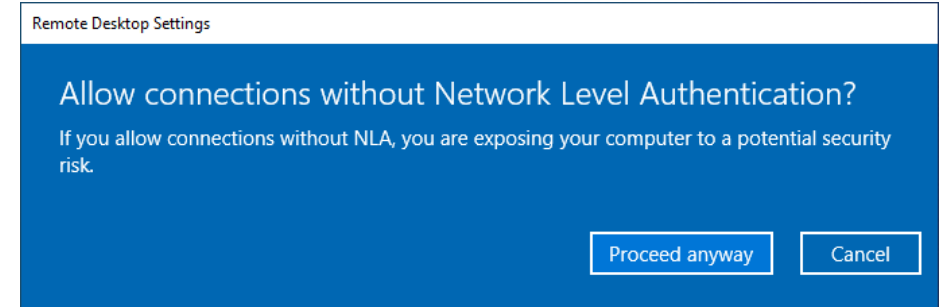
NLA : Protocole CredSSP

- [\[MS-CSSP\]](#): Credential Security Support Provider (CredSSP) Protocol
- [\[MS-SPNG\]](#): Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) Extension
- [\[MS-NLMP\]](#): NT LAN Manager (NTLM) Authentication Protocol
- [\[MS-KILE\]](#): Kerberos Protocol Extensions

Bien que les spécifications de protocoles soient destinées aux développeurs, elles peuvent servir de référence pour mieux comprendre le fonctionnement interne par les administrateurs système.

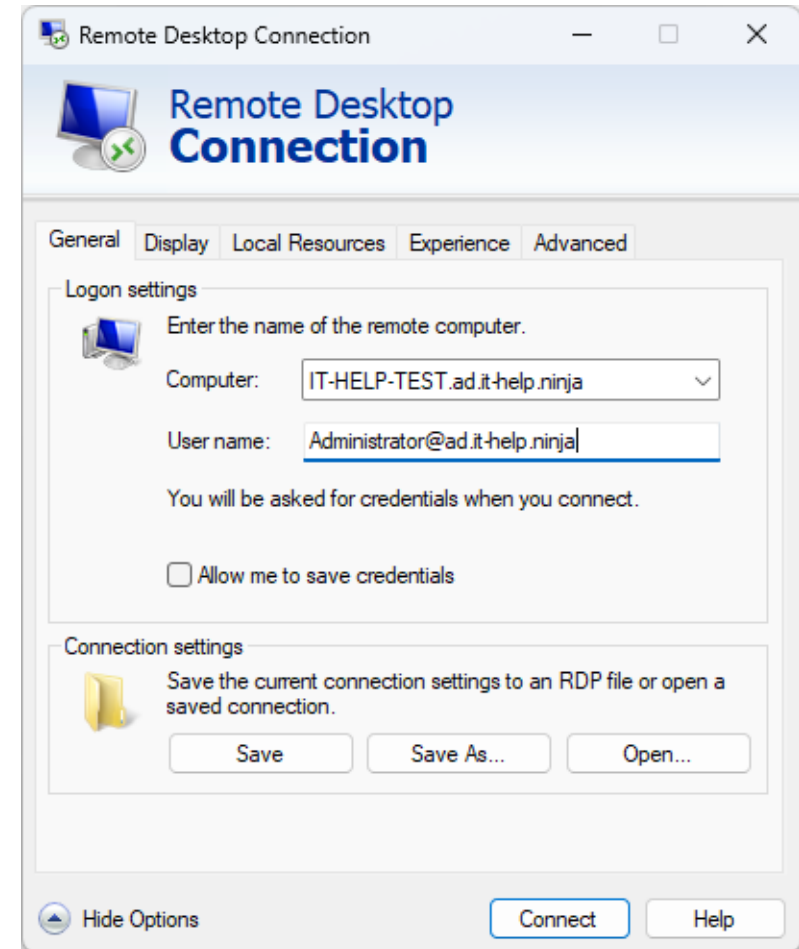
Désactiver NLA côté serveur

- Dans les paramètres *Remote Desktop* du serveur, décochez l'option **Require computers to use Network Level Authentication**.
- Ignorez le message d'avertissement pour procéder.
- Le serveur acceptera encore NLA si le client le demande.
 - Par contre, il acceptera aussi les connexions sans NLA.

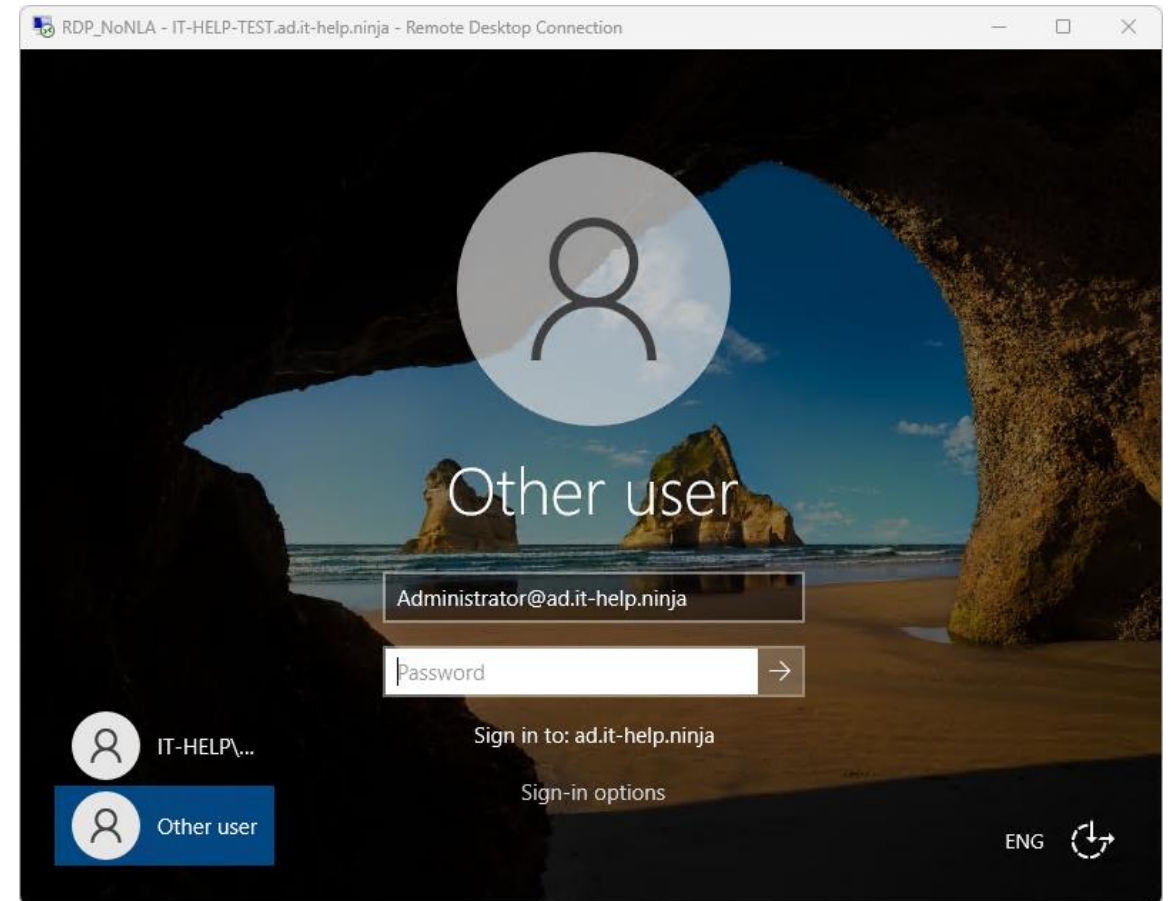
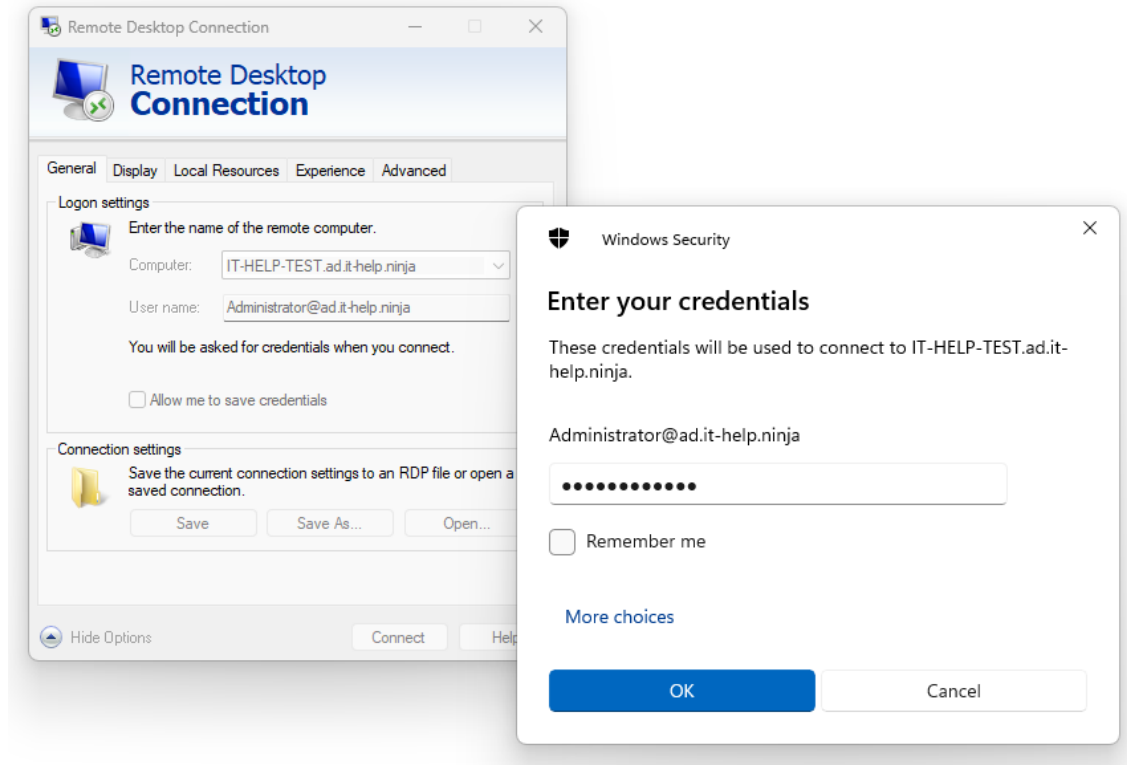


Désactiver NLA côté client

- Lancez mstsc, puis cliquez sur **Show Options**.
- Cliquez sur **Save As...**, choisissez un nom de fichier comme RDP_NoNLA.rdp.
- Ouvrez RDP_NoNLA.rdp dans Notepad.
- Ajoutez ou modifiez la ligne **enablecredsspsupport:i:0**
- Ouvrez RDP_NoNLA.rdp avec mstsc pour lancer la connexion.



RDP avec NLA et sans NLA



Couche de transport RDP UDP

- Extension RDP multi-transport
- Désactivation du transport UDP

Extension RDP multi-transport

RDP utilise toujours une connexion TCP avant d'ouvrir une connexion UDP.
C'est l'extension de protocole RDP multi-transport ([\[MS-RDPENT\]](#)).

Dans Wireshark, faites attention : utilisez **tcp.port == 3389 || udp.port == 3389**

L'ouverture du transport UDP est optionnelle, mais peut facilement causer problème :

- Lors de l'utilisation d'un VPN
- Avec une carte réseau un peu capricieuse
- Avec une mise à jour Windows qui brise RDP UDP

Désactivation du transport UDP

En cas de doute, désactivez la couche de transport UDP dans le client RDP :

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal  
Services\Client' -Name 'fClientDisableUDP' -Value 1
```

Le problème n'est peut-être pas l'authentification, mais un problème de couche de transport UDP causant des erreurs intermittentes lors de l'ouverture de la connexion.

Configuration TLS dans RDP

- Comment forcer la désactivation TLS
- Symptômes de l'absence de TLS côté client
- Erreur de validation de certificat avec TLS
- Exception de validation de certificat avec TLS
- Erreur de validation d'état de révocation
- Importance de TLS dans RDP
- Configurer un certificat TLS dans RDP
- Certificats TLS RDP avec Active Directory
- Création du modèle de certificat RDP
- Diagnostiquer la sélection du certificat RDP

Comment forcer la désactivation TLS

- TLS peut parfois se désactiver par erreur dans RDP
 - Il n'est pas très clair pourquoi, mais ça peut arriver
- Pour simuler le problème, lancez gpedit.msc :
 - **Computer Configuration -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Security**
 - **Require use of specific security layer for remote (RDP) connections**
 - Activez et sélectionnez la valeur « RDP » pour **Security Layer**.

Local Group Policy Editor

File Action View Help

NetMeeting
OneDrive
Online Assistance
OOBE
Portable Operating System
Presentation Settings
Push To Install
Remote Desktop Services
RD Licensing
Remote Desktop Connection Client
Remote Desktop Session Host
Application Compatibility
Connections
Device and Resource Redirection
Licensing
Printer Redirection
Profiles
RD Connection Broker
Remote Session Environment
Security
Session Time Limits
Temporary folders

Security

Require use of specific security layer for remote (RDP) connections

Edit [policy setting](#)

Requirements:
At least Windows Vista

Description:
This policy setting specifies whether to require the use of a specific security layer to secure communications between clients and RD Session Host servers during Remote Desktop Protocol (RDP) connections.

If you enable this policy setting, all communications between clients and RD Session Host servers during remote connections must use the security method specified in this setting. The following security methods are available:

Setting	State
Server authentication certificate template	Not configured
Set client connection encryption level	Not configured
Always prompt for password upon connection	Not configured
Require secure RPC communication	Not configured
Require use of specific security layer for remote (RDP) conn...	Enabled
Do not allow local administrators to customize permissions	Not configured
Require user authentication for remote connections by usin...	Not configured

Require use of specific security layer for remote (RDP) connections

Require use of specific security layer for remote (RDP) connections

Previous Setting Next Setting

☐ Not Configured
☒ Enabled
☐ Disabled

Comment:

Supported on: At least Windows Vista

Options:

Security Layer: RDP
Choose the security layer from the drop-down list.

Help:

This policy setting specifies whether to require the use of a specific security layer to secure communications between clients and RD Session Host servers during Remote Desktop Protocol (RDP) connections.

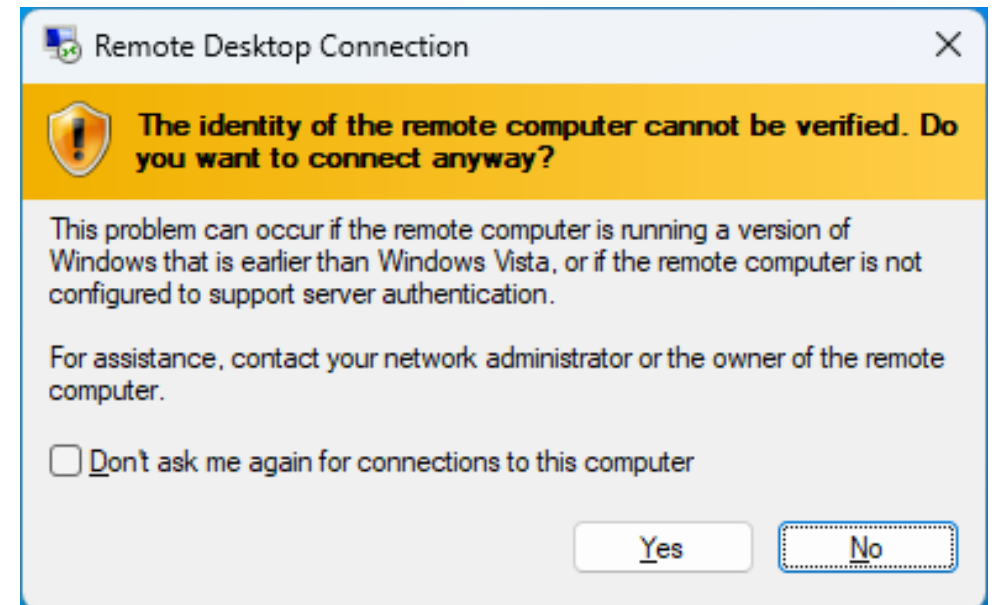
If you enable this policy setting, all communications between clients and RD Session Host servers during remote connections must use the security method specified in this setting. The following security methods are available:

7 setting(s)

Symptômes de l'absence de TLS côté client

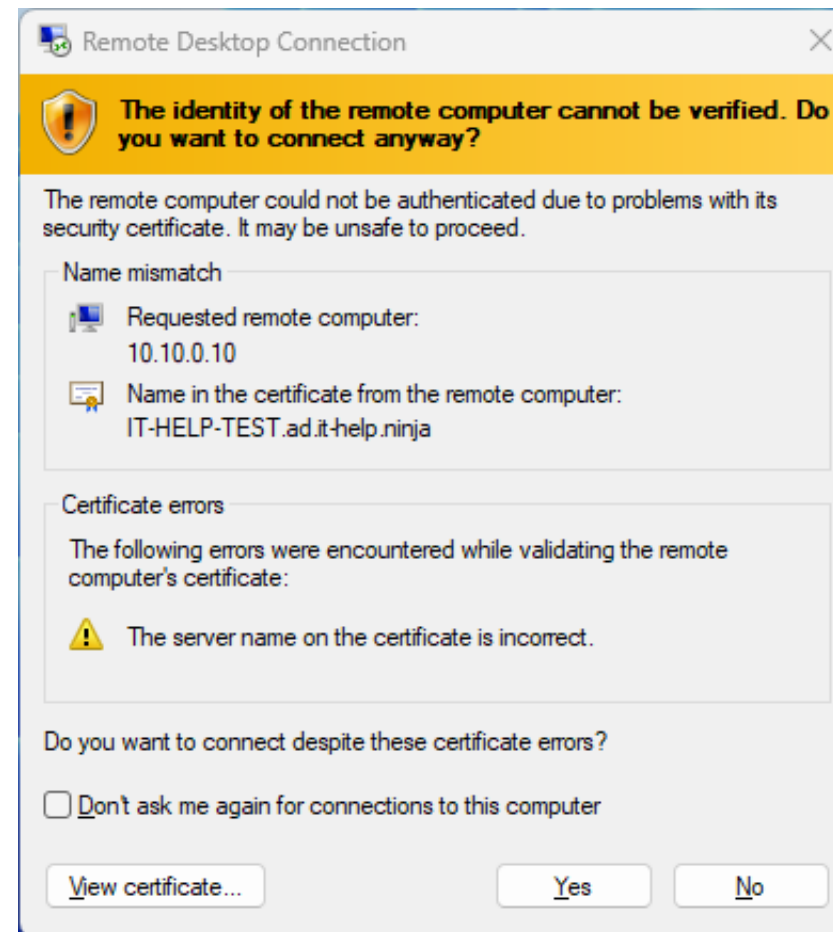
- Au lieu d'une erreur de validation de certificat, on obtient une fenêtre d'avertissement jaune faisant la mention de Windows Vista.
- Ce message passe la plupart du temps inaperçu étant donné l'habitude très ancrée chez les utilisateurs d'ignorer les messages d'avertissement RDP.
- L'absence de TLS est une situation très anormale.

Investiguez et corrigez la situation immédiatement.



Erreur de validation de certificat avec TLS

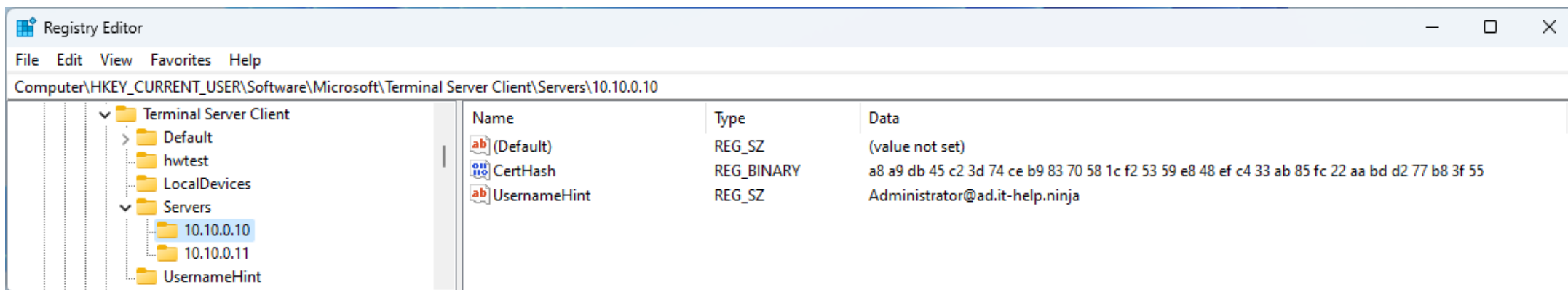
- Si vous avez coché « **Don't ask me again for connections to this computer** », supprimez les clés de registre sous HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Servers.
- Si vos certificats RDP se valident automatiquement, forcez une erreur de validation en utilisant l'adresse IP.
- Remarquez comment le message d'avertissement jaune fait mention d'un certificat lorsque TLS est utilisé.



Exception de validation de certificat avec TLS

Cocher « **Don't ask again for connections to this computer** » enregistre le hash SHA1 du certificat sous **HKCU:\Software\Microsoft\Terminal Server Client\Servers\<ServerName>\CertHash (REG_BINARY)**.

UsernameHint enregistre le dernier nom d'utilisateur pour le serveur.



Erreur de validation d'état de révocation

Un problème courant est la validation de l'état de révocation

- La cause fréquente est un fichier CRL expiré dans le cache local.
- L'URL de distribution du CRL est souvent inaccessible ou bloquée.

Pour contourner le problème:

```
New-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Lsa\Credssp" -Name  
UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors -Value 1 -Force
```

Pour invalider le cache CRL:

```
& certutil.exe "-urlcache" "crl" "delete"  
& certutil.exe "-setreg" "chain\ChainCacheResyncFiletime" "@now"
```

Importance de TLS dans RDP

- **TLS devrait toujours être utilisé dans RDP, combiné avec NLA.**
 - La couche de transport archaïque n'a aucune raison d'être utilisée aujourd'hui.
- La validation des certificats est **recommandée**, mais apporte peu de valeur réelle.
 - L'échange NLA inclut un mécanisme anti-MITM basé sur l'authentification Windows.
 - Le client RDP Microsoft valide le certificat **après** avoir débuté l'authentification (!)
 - L'habitude d'ignorer les certificats dans RDP est très ancrée parmi les utilisateurs.

En résumé, il faut toujours utiliser RDP NLA (avec TLS) et idéalement déployer des certificats automatiquement validables pour suivre les bonnes pratiques, mais les gains sont marginaux.

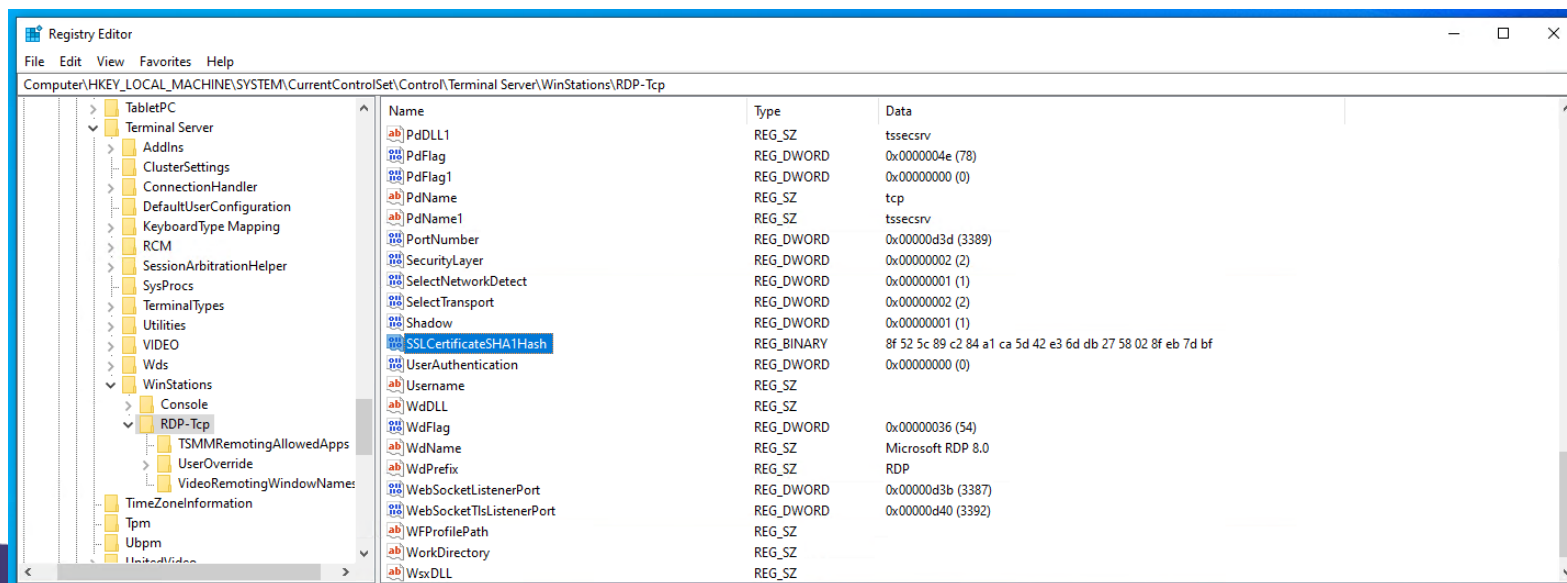
Si votre cas d'utilisation demande de désactiver RDP NLA, il est très important de compenser par une validation automatique des certificats TLS, sans quoi une attaque MITM serait facilement réalisable.

Configurer un certificat TLS dans RDP

Obtenez le hash SHA1 du certificat dans le *certificate store* de Windows, puis utilisez PowerShell :

```
Get-CimInstance -ClassName Win32_TSGeneralSetting -Namespace ROOT\CIMV2\TerminalServices |  
Set-CimInstance -Property @{ SSLCertificateSHA1Hash = $RdpCertificateThumbprint }
```

Dans le registre Windows, la propriété **SSLCertificateSHA1Hash** se trouve sous
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp



Certificats TLS RDP avec Active Directory

Dans un environnement Active Directory avec *Certificate Services*, il est possible de déployer des certificats TLS pour RDP automatiquement par une stratégie de groupe.

La sélection du certificat se fait à l'aide du nom du template AD CS utilisé pour générer le certificat

Step-by-Step Procedure to Deploy RDP Certificates Using GPO

<https://medium.com/theseckmaster/step-by-step-procedure-to-deploy-rdp-certificates-using-gpo-ba27f1c9b5fa>

Assurez-vous d'utiliser le *certificate template name* et non le *certificate display name*.

Supprimez la valeur de registre **SSLCertificateSHA1Hash** si vous l'avez configuré manuellement.

Création du modèle de certificat RDP (1/2)

- Lancez **certtmpl.msc** (*Certificate Templates Console*)
- Sélectionnez **Computer**, puis **Duplicate Template**.
- Dans l'onglet **General**, entrez **Remote Desktop** comme *display name*
 - Le *template name* deviendra **RemoteDesktop** sans les espaces
 - Attention : il faut utiliser le *template name* pour la stratégie de groupe (GPO)
- Dans l'onglet **Extensions**, sélectionnez **Application Policies**, puis **Edit**.
 - Supprimez **Client Authentication** de la liste, puis cliquez sur **Add**
 - Cliquez sur **New**, puis entrez **Remote Desktop Authentication** (1.3.6.1.4.1.311.54.1.2)
- Finalisez la création du modèle de certificat RDP

Création du modèle de certificat RDP (2/2)

- Lancez **certsrv.msc** (*Certification Authority Console*)
- Sélectionnez **Certificate Templates -> New** et ensuite **Certificate Template to Issue**
- À partir de la liste des modèles de certificat, sélectionnez **Remote Desktop**.
- Cliquez sur **OK**.

Properties of New Template

Subject Name		Server		Issuance Requirements	
Superseded Templates		Extensions		Security	
Compatibility	General	Request Handling	Cryptography	Key Attestation	

Template display name:
Remote Desktop

Template name:
RemoteDesktop

Validity period: 1 years
Renewal period: 6 weeks

☐ Publish certificate in Active Directory
☐ Do not automatically reenroll if a duplicate certificate exists in Active Directory

OK Cancel Apply Help

Remote Desktop Properties

Subject Name		Issuance Requirements	
General		Compatibility	
Request Handling		Cryptography	
Key Attestation		Superseded Templates	
Extensions		Security	
Server			

To modify an extension, select it, and then click Edit.

Extensions included in this template:

- Application Policies
- Basic Constraints
- Certificate Template Information
- Issuance Policies
- Key Usage

Edit...

Description of Application Policies:

Server Authentication
Remote Desktop Authentication

OK Cancel Apply Help

Edit Application Policies Extension

An application policy defines how a certificate can be used.

Application policies:

- Remote Desktop Authentication
- Server Authentication

Add... Edit... Remove

☐ Make this extension critical

OK Cancel

Edit Application Policy

Type the new name for this policy.

Name:
Remote Desktop Authentication

Object identifier:
1.3.6.1.4.1.311.54.1.2

OK Cancel

Group Policy Management Editor

File Action View Help

Location and Sensors
Maintenance Scheduler
Maps
MDM
Messaging
Microsoft account
Microsoft Defender Antivirus
Microsoft Defender Exploit Guard
Microsoft Secondary Authentication Factor
Microsoft User Experience Virtualization
NetMeeting
OneDrive
Online Assistance
OOBE
Portable Operating System
Presentation Settings
Push To Install
Remote Desktop Services
RD Licensing
Remote Desktop Connection Client
Remote Desktop Session Host
Application Compatibility
Connections
Device and Resource Redirection
Licensing
Printer Redirection
Profiles
RD Connection Broker
Remote Session Environment
Security

Security

Server authentication certificate template

Edit [policy setting](#)

Requirements:
At least Windows Vista

Description:
This policy setting allows you to specify the name of the certificate template that determines which certificate is automatically selected to authenticate an RD Session Host server.

A certificate is needed to authenticate an RD Session Host server when TLS 1.0, 1.1 or 1.2 is used to secure communication between a client and an RD Session Host server during RDP connections.

If you enable this policy setting, you need to specify a certificate template name. Only certificates created by using the specified certificate template will be considered when a certificate to authenticate the RD Session Host server is automatically selected. Automatic certificate selection only occurs when a specific

Setting	State	Comment
Server authentication certificate template	Enabled	No
Set client connection encryption level	Not configured	No
Always prompt for password upon connection	Not configured	No
Require secure RPC communication	Not configured	No
Require use of specific security layer	Not configured	No
Do not allow local administrators to	Not configured	No
Require user authentication for rem	Not configured	No

Server authentication certificate template

Server authentication certificate template

Previous Setting

☐ Not Configured Comment:

☒ Enabled

☐ Disabled

Supported on: At least Windows Vista

Options:

Certificate Template Name

RemoteDesktop

Help:

This policy setting allows you to certificate template that determin automatically selected to auther

A certificate is needed to auther when TLS 1.0, 1.1 or 1.2 is used t between a client and an RD Sess connections.

If you enable this policy setting, template name. Only certificate:

Diagnostiquer la sélection du certificat RDP

Quel certificat a été sélectionné?

Utilisez Wireshark pour inspecter le certificat dans le handshake TLS 1.2!

RDP ne prend pas encore en charge TLS 1.3, donc une bonne partie de l'information pertinente est lisible.

The image shows a Wireshark capture of an RDP connection. The top pane displays the packet list, and the bottom pane shows the details of the selected packet (No. 83, Time 6.724982).

Packet List:

No.	Time	Source	Destination	Protocol	Length	Info
10	2.616062	10.10.0.6	10.10.0.1	TCP	66	3389 → 30691 [SYN, ACK] Seq=0 Ack=1 Win=64000 Len=0 MSS=1460 WS=1 SACK_PERM
11	2.616146	10.10.0.1	10.10.0.6	TCP	54	30691 → 3389 [ACK] Seq=1 Ack=1 Win=262656 Len=0
12	2.616915	10.10.0.1	10.10.0.6	RDP	101	Cookie: msthash=IT-HELP\A, Negotiate Request
13	2.636764	10.10.0.6	10.10.0.1	TCP	54	3389 → 30691 [ACK] Seq=1 Ack=48 Win=63953 Len=0
14	2.667514	10.10.0.6	10.10.0.1	RDP	73	Negotiate Response
15	2.718587	10.10.0.1	10.10.0.6	TCP	54	30691 → 3389 [ACK] Seq=48 Ack=20 Win=262656 Len=0
81	6.720952	10.10.0.1	10.10.0.6	TLSv1.2	343	Client Hello (SNI=IT-HELP-DVLS.ad.it-help.ninja)
82	6.724976	10.10.0.6	10.10.0.1	TCP	1514	3389 → 30691 [ACK] Seq=20 Ack=337 Win=63664 Len=1460 [TCP segment of a reassembled P...
83	6.724982	10.10.0.6	10.10.0.1	TLSv1.2	238	Server Hello, Certificate, Server Key Exchange, Server Hello Done
84	6.725045	10.10.0.1	10.10.0.6	TCP	54	30691 → 3389 [ACK] Seq=337 Ack=1664 Win=262656 Len=0

Details of Packet 83 (Server Hello, Certificate, Server Key Exchange, Server Hello Done):

- signedCertificate
 - version: v3 (2)
 - serialNumber: 0x3100000011dd993c4c2d02ad6100000000011
 - signature (sha256WithRSAEncryption)
 - Algorithm Id: 1.2.840.113549.1.1.11 (sha256WithRSAEncryption)
 - issuer: rdnSequence (0)
 - rdnSequence: 4 items (id-at-commonName=IT-HELP-DC,dc=ad,dc=it-help,dc=ninja)
 - RDNSequence item: 1 item (dc=ninja)
 - RDNSequence item: 1 item (dc=it-help)
 - RDNSequence item: 1 item (dc=ad)
 - RDNSequence item: 1 item (id-at-commonName=IT-HELP-DC)
 - validity
 - notBefore: utcTime (0)
 - utcTime: 2024-04-29 19:03:51 (UTC)
 - notAfter: utcTime (0)
 - utcTime: 2025-04-29 19:03:51 (UTC)
 - subject: rdnSequence (0)
 - rdnSequence: 0 items
 - subjectPublicKeyInfo
 - algorithm (rsaEncryption)
 - Algorithm Id: 1.2.840.113549.1.1.1 (rsaEncryption)
 - subjectPublicKey [truncated]: 3082010a0282010100de8f63978deec16e0da192a205188400546c9d7...
 - modulus: 0x00de8f63978deec16e0da192a205188400546c9d7d4daae4c8a80a16d70d67120ecc552b...
 - publicExponent: 65537
 - extensions: 9 items
 - Extension (id-ms-certificate-template)
 - Extension Id: 1.3.6.1.4.1.311.21.7 (id-ms-certificate-template)
 - CertificateTemplate
 - templateID: 1.3.6.1.4.1.311.21.8.702800.9003539.9061025.16416222.10900147.24.1044;
 - templateMajorVersion: 100
 - templateMinorVersion: 3

RDP NLA sans délégation

- Mode *Restricted Admin* (RA)
- Remote Credential Guard (RCG)

Mode *Restricted Admin* (RA)

Le mode *Restricted Admin* (RA) saute l'étape de délégation des identifiants dans CredSSP.

- On obtient un *network logon* au lieu d'un *remote interactive logon*.
- Les identifiants ne sont jamais envoyés au serveur, hors de portée pour mimikatz.
- Opérations dans la session Windows limitées par le *network logon* (pas de double hop).
- Le mode *Restricted Admin* doit être activé sur le serveur et demandé par le client.

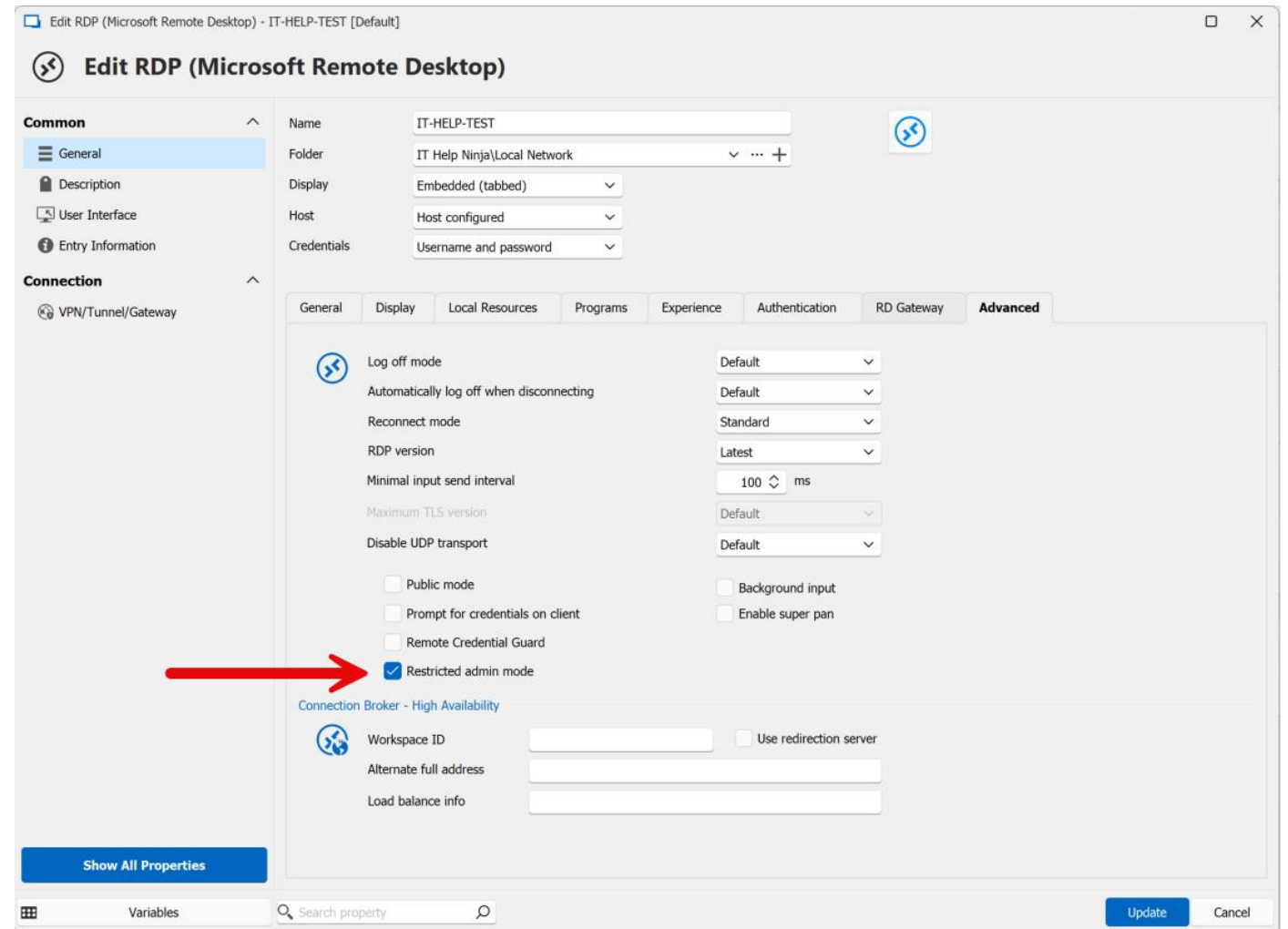
Côté serveur RDP, activez le mode *Restricted Admin* (RA) :

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Lsa" -Name  
"DisableRestrictedAdmin" -Value 0
```

Coté client RDP, lancez mstsc avec l'option /restrictedAdmin.

Restricted Admin dans Remote Desktop Manager

Le mode *Restricted Admin* est disponible dans **Remote Desktop Manager**.



Inconvénients du Mode *Restricted Admin*

- Il est désactivé par défaut sur le serveur RDP.
- Il n'est pas possible de forcer le client RDP à l'utiliser.
- L'étape de délégation en moins, on ouvre la porte aux attaques de type *pass-the-hash*.
 - FreeRDP a même une option */pth* (*pass-the-hash*) acceptant un hash NTLM.
 - Les guides de tests d'intrusion traitent souvent de *Restricted Admin* + *pass-the-hash*.
- L'absence de *remote interactive logon* est trop limitative dans la session RDP.

En résumé, le mode *Restricted Admin* est une fausse bonne idée de Microsoft pour contrer le *credential grabbing* par mimikatz sur un serveur RDP potentiellement compromis.

Remote Credential Guard (RCG)

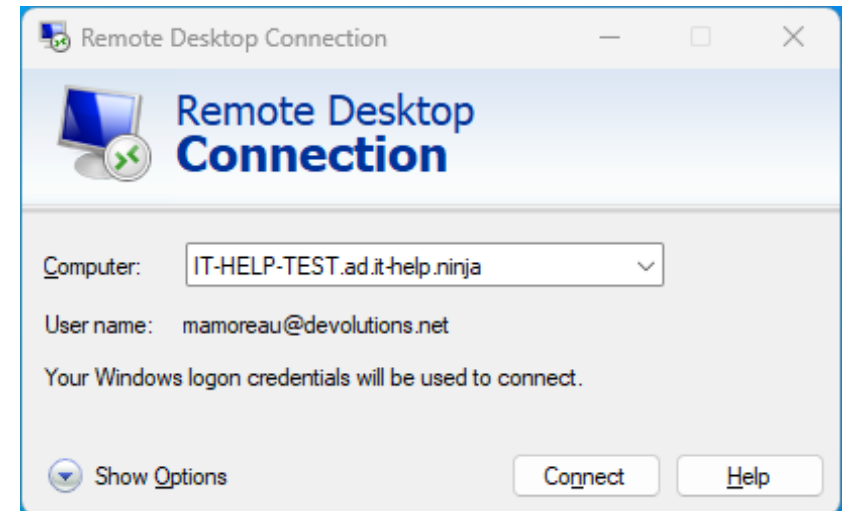
Remote Credential Guard (RCG) est une variante du mode *Restricted Admin* (RA) utilisant un canal virtuel RDP pour rediriger les appels du *Local Security Authority* (LSA) **pour obtenir un *remote interactive logon* sans jamais envoyer les identifiants au serveur.**

Côté serveur RDP, activez Remote Credential Guard (RCG) avec la même clé de registre que le mode *Restricted Admin* (RA) :

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Lsa" -Name  
"DisableRestrictedAdmin" -Value 0
```

Côté client RDP, lancez mstsc avec l'option /remoteGuard.
Il n'est pas possible de fournir des identifiants explicitement (!)

« **Your Windows logon credentials will be used to connect.** »



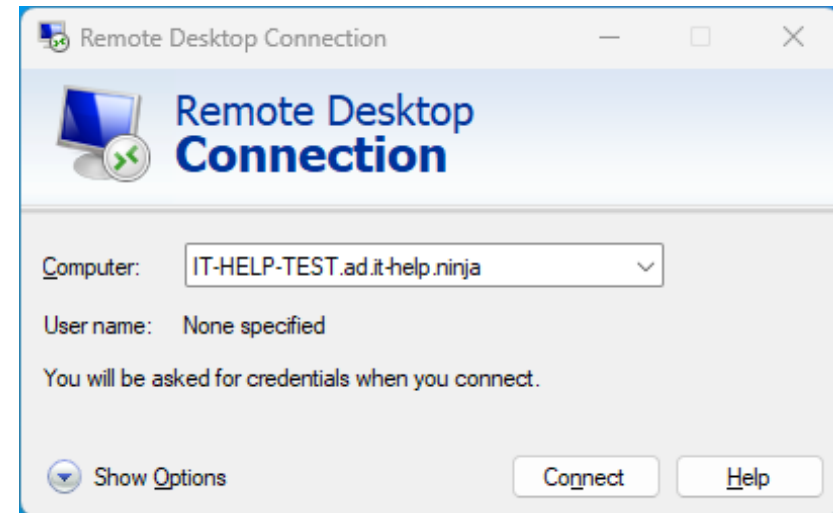
Identifiants fournis à Remote Credential Guard

L'implémentation de Remote Credential Guard présume (à tort) que l'administrateur se connecte à d'autres systèmes avec les mêmes identifiants que la session Windows sur le poste client.

Il est cependant possible de forcer mstsc à demander les identifiants avec RCG :

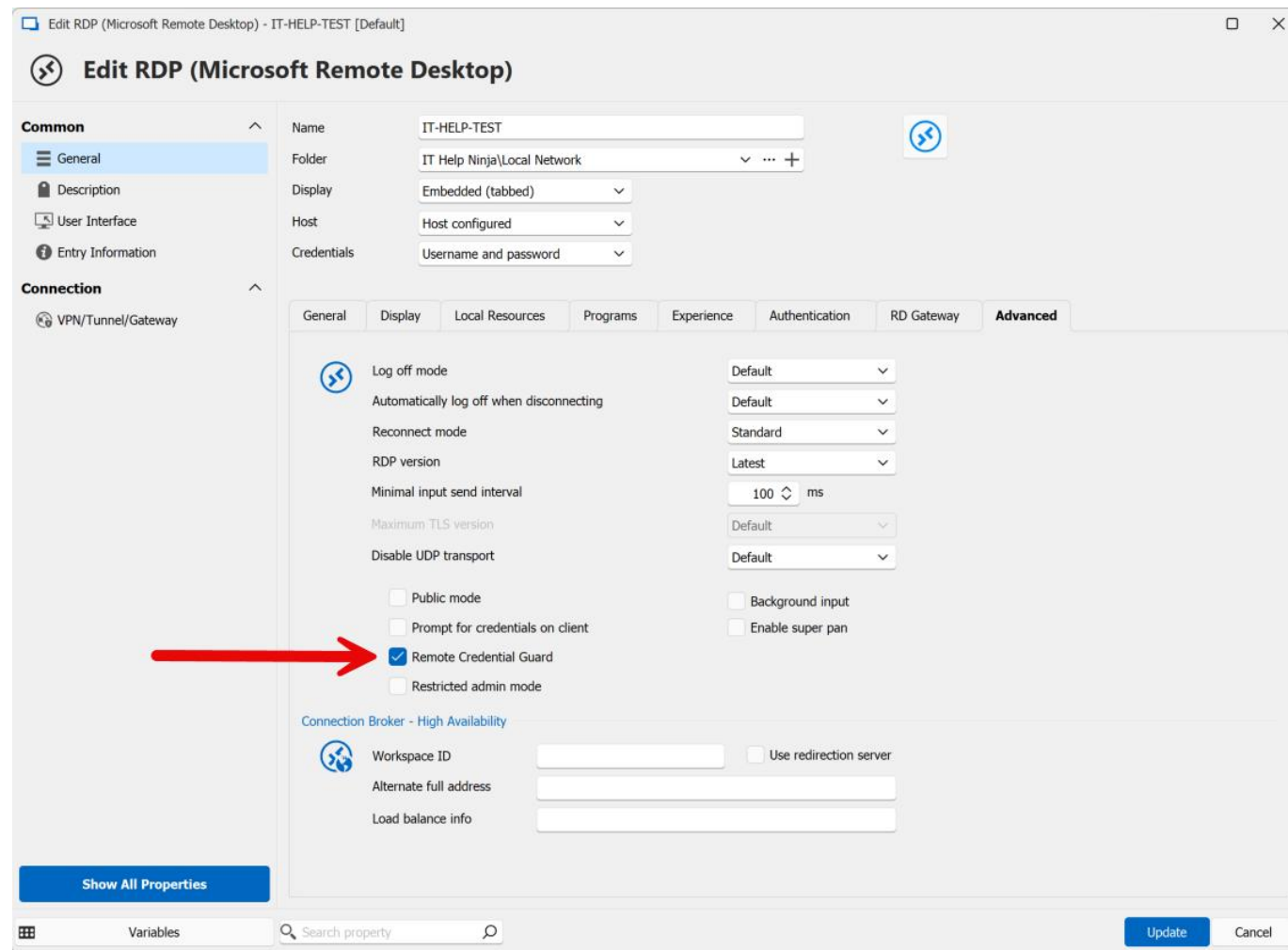
mstsc /remoteGuard /prompt

« You will be asked for credentials when you connect. »



RCG dans Remote Desktop Manager

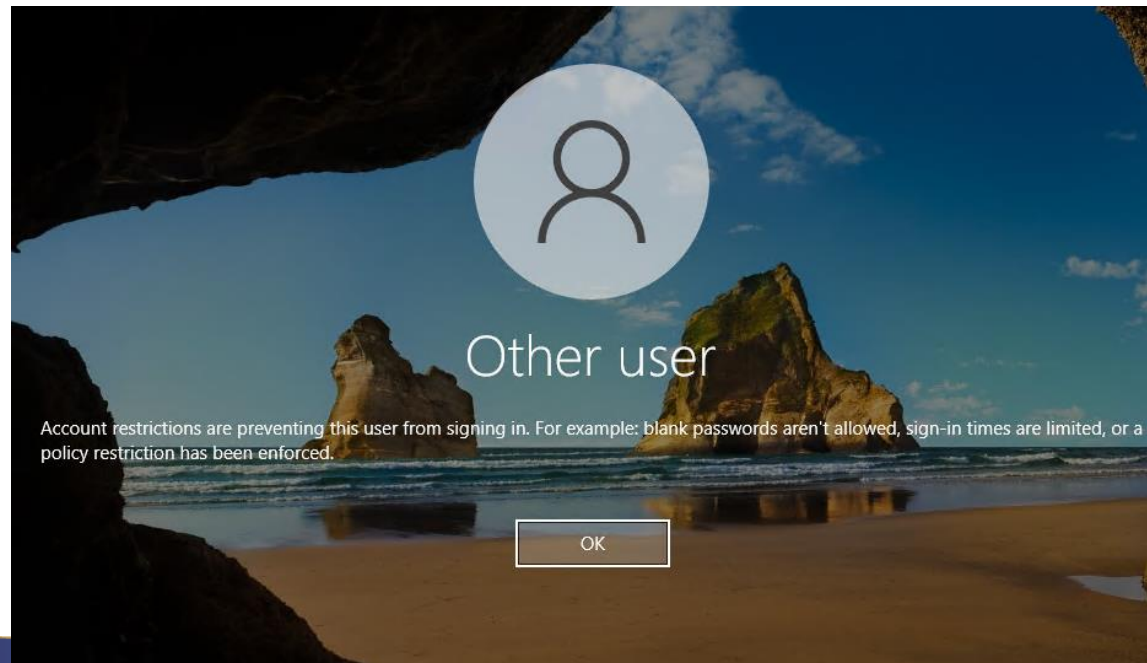
Remote Desktop Manager prend en charge Remote Credential Guard avec l'injection d'identifiants fournis explicitement, ce que mstsc ne fait pas normalement.



Symptômes de RCG désactivé coté serveur

Lorsque Remote Credential Guard (RCG) ou le mode *Restricted Admin* (RA) est demandé par le client sur un serveur RDP ne l'ayant pas activé, l'erreur suivante apparaît dans Winlogon :

« Account restrictions are preventing this user from signing in. For example: blank passwords aren't allowed, sign-in times are limited, or a policy restriction has been enforced. »



RCG et Windows Hello for Business

Remote Credential Guard a été conçu dans l'unique but de prévenir le *credential grabbing* par mimikatz sur le serveur RDP tout en permettant un *remote interactive logon*.

Cependant, en redirigeant les appels du *Local Security Authority* (LSA) par un canal virtuel RDP, Remote Credential Guard supporte **accidentellement** l'authentification sans mot de passe.

Le protocole CredSSP prévoit seulement l'authentification par mot de passe ou par carte à puce.

En l'absence d'une meilleure solution, Remote Credential Guard peut donc être utilisé dans certains cas pour prendre en charge l'authentification sans mot de passe de Windows Hello for Business.

Inconvénients de Remote Credential Guard

- Mêmes inconvénients que le mode *Restricted Admin* (RA)
 - Désactivé par défaut et impossible d'en forcer l'utilisation à partir du serveur
 - Il est cependant possible de forcer le client RDP à l'utiliser
- Support officiel limité à mstsc sur Windows seulement
- Inutilisable en pratique entre machines de domaines distincts
- Utilisable de façon aléatoire avec identifiants fournis explicitement

En résumé, même si le code existe pour utiliser des identifiants fournis explicitement dans le client RDP de Microsoft, c'est un cas d'utilisation très peu testé avec un comportement difficile à prédire.

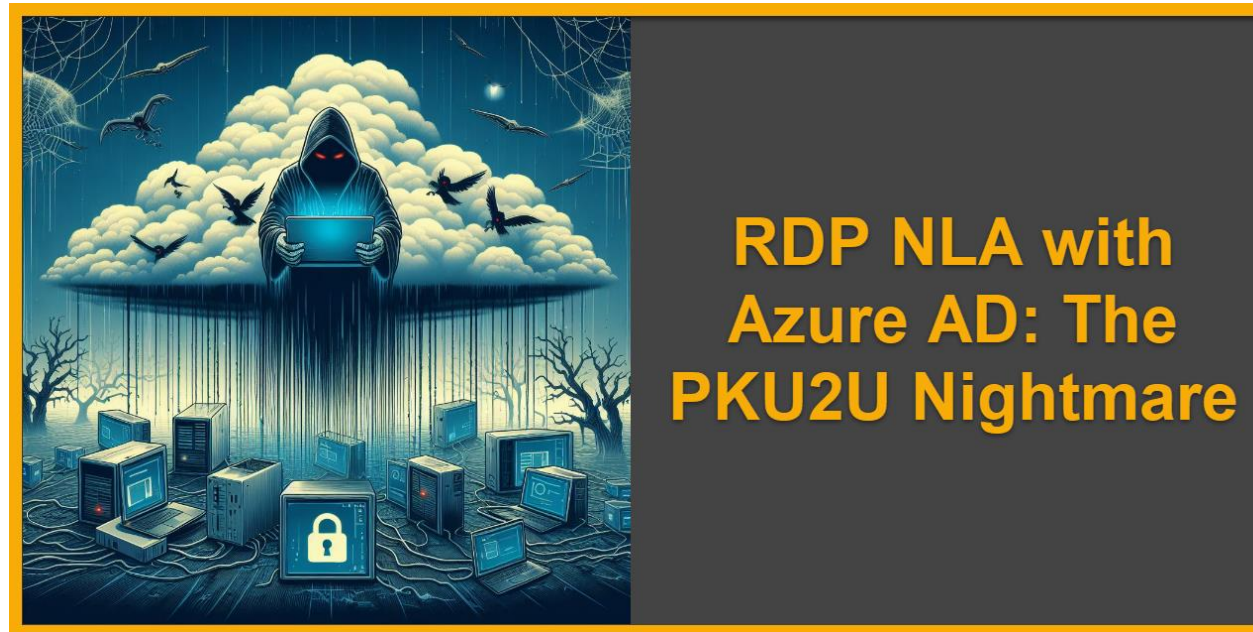
RDP Azure AD / Entra ID

- RDP NLA + Entra ID (PKU2U)
- SSO d'Entra ID pour RDP (RDS AAD)

RDP NLA + Entra ID (PKU2U)

- RDP NLA avec un compte Entra ID et le protocole PKU2U est très pénible à opérer.
 - Pas de panique : j'ai écrit un blogue détaillé sur le sujet!

<https://awakecoding.com/posts/rdp-nla-with-azure-ad-the-pku2u-nightmare/>



Contraintes RDP NLA + Entra ID (PKU2U)

- PKU2U doit être activé côté client et serveur (désactivé par défaut)
- Le client doit être *workplace joined* avec le même compte Entra ID
 - Même *tenant* Entra ID, mais aussi le même compte Entra ID dans Windows
 - Windows permet d'être authentifié seulement à 5 comptes Entra ID à la fois
- Protocole documenté partiellement – la portion critique avec Azure est absente
 - Support tierce-partie limité sur Windows et inexistant sur les autres plateformes
 - Dépendance sur le cache du *Primary Refresh Token* (PRT) de Windows pour PKU2U

PKU2U est un protocole originalement conçu pour les groupes de travail Windows qui a été récupéré comme solution « temporaire » dans RDP NLA + Entra ID en attendant le SSO d'Entra ID (RDS AAD auth).

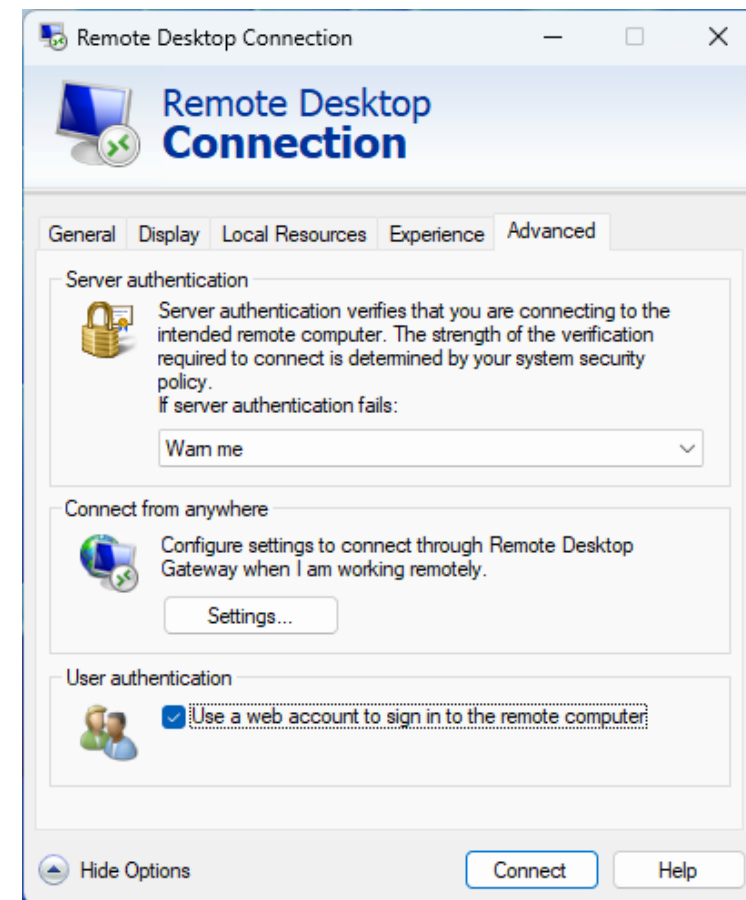
SSO d'Entra ID pour RDP (RDS AAD Auth)

RDP avec un compte Entra ID est possible avec le SSO d'Entra ID!

- Fenêtre de connexion Web avec Azure dans mstsc
- RDP MFA supporté nativement avec FIDO2 / WebAuthn
- Utilise un jeton spécial au lieu de CredSSP pour s'authentifier
- Une fois autorisé pour une machine, le SSO reste valide plusieurs mois
- Solution de remplacement à l'ancien protocole RDP NLA + PKU2U

L'option est `enablerdsaaauth:i:1` dans le fichier .RDP.

« **Use a web account to sign in to the remote computer** » dans mstsc



Inconvénients du SSO d'Entra ID pour RDP

- **Il n'est pas possible d'invalider l'autorisation à partir du poste client**
 - Je répète : une fois autorisé, le client reste autorisé plusieurs mois!
 - Il faut invalider manuellement l'autorisation par le *tenant* Entra ID.
- **Intégration tierce-partie très limitée**
 - Pas d'injection d'identifiants avec mstsc ou l'ActiveX RDP sur Windows.
 - Le seul client_id OAuth utilisable pour le SSO d'Entra ID est bloqué à mstsc.

RDP NLA avec Kerberos

- Survol de Kerberos
- Configuration DNS
- Détection du KDC
- Kerberos sur macOS

Pourquoi Kerberos est si difficile?

- Parce que le downgrade NTLM fonctionne silencieusement
- Parce que Kerberos communique hors bande avec le KDC
- Parce que le client n'est pas configuré pour trouver le KDC
- Parce que le client ne peut pas trouver ni rejoindre le KDC
- Parce que le client n'utilise pas le module SSPI Negotiate
- Parce que le client ne prend tout simplement pas Kerberos en charge
- Parce que le client n'est pas Windows et krb5.conf n'est pas configuré
- Parce que le client est Android ou iOS, sans GSSAPI ni krb5.conf
- Parce que c'est trop tentant de se connecter par l'adresse IP en RDP
- Parce que le DNS est mal configuré, ou qu'on ne veut pas changer le DNS
- Parce que c'est juste trop pénible à diagnostiquer, donc on laisse faire
- Parce que vous avez [désactivé le service WPAD sans savoir ce que ça brise](#)
- Parce que vous avez essayé le [KDC proxy](#) et c'est une boîte noire à problèmes
- Parce que Microsoft est responsable d'une bonne partie de l'utilisation NTLM
- Parce que ça l'air trop compliqué et que vous avez d'autres choses à faire 😊



Jonathan
@jrog404

...

Tried putting admins in the protected users group today and immediately broke RDP. How's your day?

4:53 PM · Apr 22, 2024 · 5,464 Views

NTLM, le talon d'Achille de Windows

NTLM est vulnérable à plusieurs types d'attaques

- Attaque par relai de jetons NTLM entre protocoles (SMB vers LDAP, etc)
- Attaque par *NTLM responder* pour récolter des hashes NetNTLM
- Attaque par récolte passive sur le réseau des hashes NetNTLM
- Attaque par extraction des hashes NTLM de la base de données SAM

NTLM est fondamentalement limité par sa conception

- Aucune validation du serveur (comme avec TLS ou Kerberos)
- Aucune authentification mutuelle ou bidirectionnelle
- Fonctionne uniquement en fonction d'un hash de mot de passe
- Fonctionne par *challenge/response* sans système de jetons
- Déprécié par Microsoft, sans aucune amélioration de sécurité

Nomenclature : DC et KDC

DC : (Active Directory) *Domain Controller*

KDC : (Kerberos) *Key Distribution Center*

Étant donné que le Kerberos KDC se trouve sur le contrôleur de domaine (DC), on utilise souvent de façon interchangeable KDC et DC lorsqu'on parle de détecter le KDC nécessaire à Kerberos.

Le contrôleur de domaine inclut le service Kerberos KDC, mais aussi les services LDAP et DNS.

Mélanger DC et KDC n'est pas si grave puisqu'on fait souvent référence à la même machine.

Article de référence: [How domain controllers are located in Windows](#)

Kerberos et Active Directory (1/2)

Kerberos est uniquement possible avec des comptes Active Directory

- Le contrôleur de domaine (DC) sert de Kerberos *Key Distribution Center* (KDC)
- Les comptes locaux n'ont pas de KDC, et utilisent donc uniquement NTLM
- Les serveurs utilisent un Kerberos *Service Principal Name* (SPN)
 - TERMSRV/IT-HELP-TEST.ad.it-help.ninja
- Les clients doivent normalement utiliser un *User Principal Name* (UPN)
 - [Administrator@ad.it-help.ninja](#) (format UPN)
 - IT-HELP/Administrator (format SAM ou *downlevel*, non recommandé)

Kerberos et Active Directory (2/2)

- Le Kerberos *realm* est la plupart du temps le suffixe DNS du domaine
 - .ad.it-help.ninja
- Kerberos est intimement lié au DNS du domaine Active Directory
 - L'utilisation du FQDN du serveur est nécessaire (IP = NTLM)
 - La détection du KDC se fait en fonction du Kerberos realm
- Oui, ça en fait beaucoup, on va y repasser là-dessus étape par étape. 😊

Comment tester que c'est bien Kerberos?

#1 : Ajouter l'utilisateur dans le groupe *Protected Users* d'Active Directory

Solution la plus simple avec impact limité par utilisateur

Désactive NTLM, durcit le chiffrement Kerberos et écourte la durée de vie des jetons

#2 : Désactiver l'authentification NTLM sortante sur le client (*outbound NTLM*)

Computer Configuration / Windows Settings / Security Settings / Local Policies / Security Options

Network security: Restrict NTLM: Outgoing NTLM traffic to remote servers: « Deny all »

#3 : Désactiver l'authentification NTLM entrante sur le serveur (*inbound NTLM*)

Computer Configuration / Windows Settings / Security Settings / Local Policies / Security Options

Network security: Restrict NTLM: Incoming NTLM traffic: « Deny all domain accounts »

#4 : Déchiffrer le trafic TLS RDP dans Wireshark pour en avoir la preuve irréfutable

Oui, c'est souvent la meilleure solution pour en être certain. (Je me suis fait avoir plus d'une fois.)

Groupe *Protected Users* d'Active Directory

The screenshot displays the Active Directory Users and Computers console. The left pane shows the tree structure with 'Users' selected. The main pane lists domain objects with columns for Name, Type, and Description. The 'Protected Users' group is highlighted. A 'Protected Users Properties' dialog box is open, showing the 'Members' tab with a single member: 'ProtectedUser' (ad.it-help.ninja/Users).

Name	Type	Description
Administrator	User	Built-in account for administering the computer/domain
Allowed RODC Password Replication...	Security Group - Domain Local	Members in this group can have their passwords replicated to all read-only domain controllers in t...
Cert Publishers	Security Group - Domain Local	Members of this group are permitted to publish certificates to the d...
Cloneable Domain Controllers	Security Group - Global	Members of this group that are domain controllers may be cloned.
Denied RODC Password Replication ...	Security Group - Domain Local	Members in this group cannot have their passwords replicated to ar...
DnsAdmins	Security Group - Domain Local	DNS Administrators Group
DnsUpdateProxy	Security Group - Global	DNS clients who are permitted to perform dynamic updates on beh...
Domain Admins	Security Group - Global	Designated administrators of the domain
Domain Computers	Security Group - Global	All workstations and servers joined to the domain
Domain Controllers	Security Group - Global	All domain controllers in the domain
Domain Guests	Security Group - Global	All domain guests
Domain Users	Security Group - Global	All domain users
Enterprise Admins	Security Group - Universal	Designated administrators of the enterprise
Enterprise Key Admins	Security Group - Universal	Members of this group can perform administrative actions on key o...
Enterprise Read-only Domain Contr...	Security Group - Universal	Members of this group are Read-Only Domain Controllers in the en...
Group Policy Creator Owners	Security Group - Global	Members in this group can modify group policy for the domain
Guest	User	Built-in account for guest access to the computer/domain
Key Admins	Security Group - Global	Members of this group can perform administrative actions on key o...
Protected Users	Security Group - Global	Members of this group are afforded additional protections against a...
ProtectedUser	User	User member of the Protected Users group

Protected Users Properties

General Members Member Of Managed By

Members:

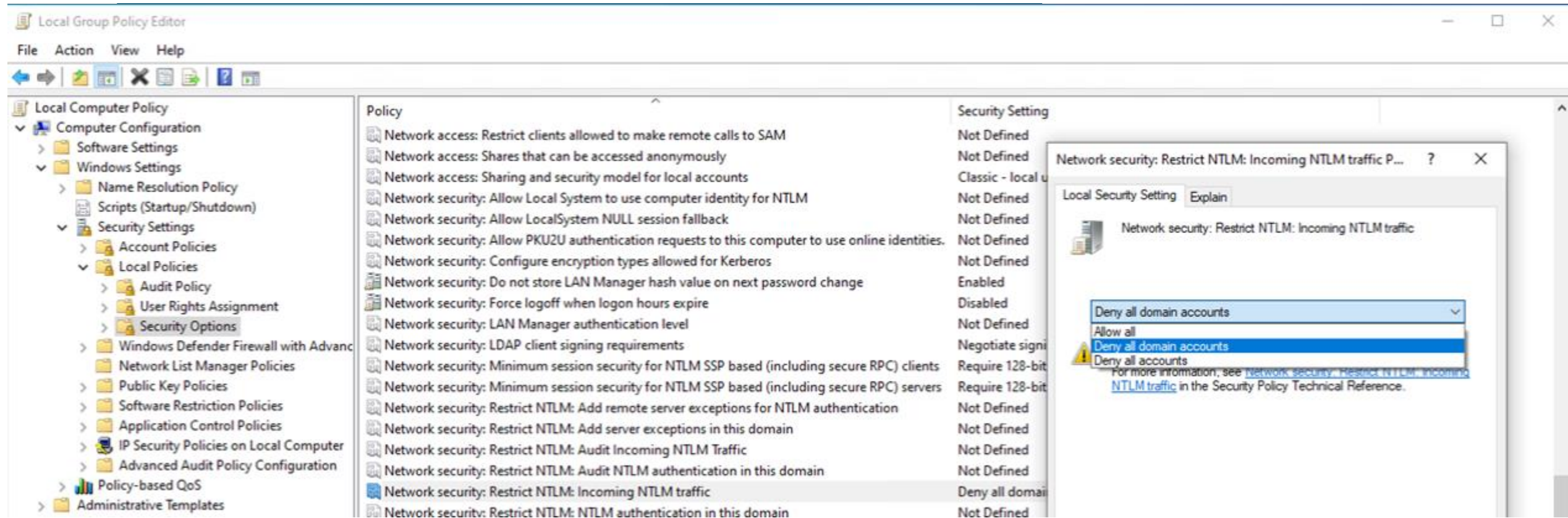
Name	Active Directory Domain Services Folder
ProtectedUser	ad.it-help.ninja/Users

Désactiver l'authentification NTLM sortante

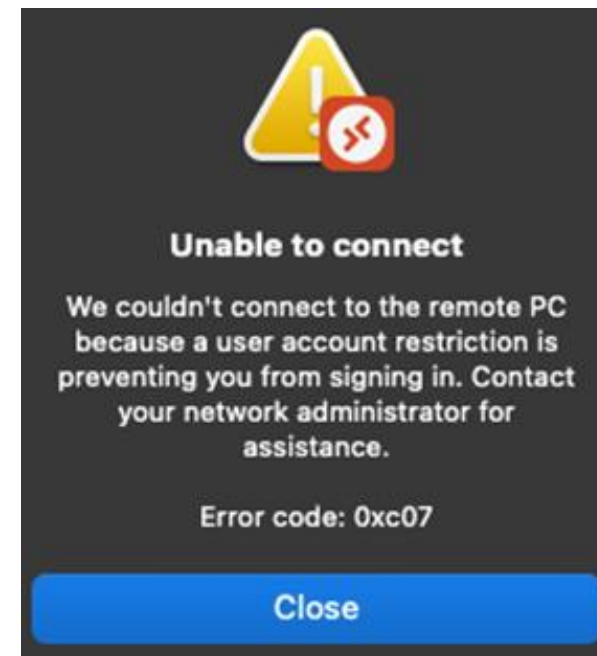
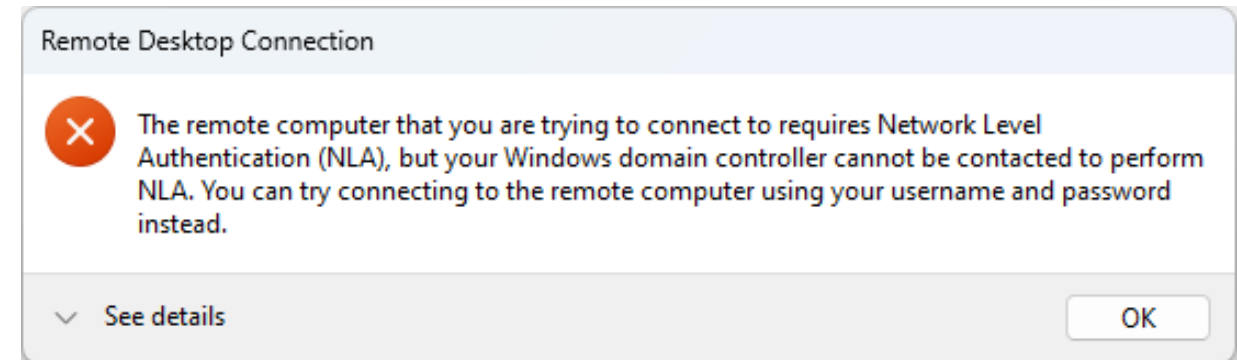
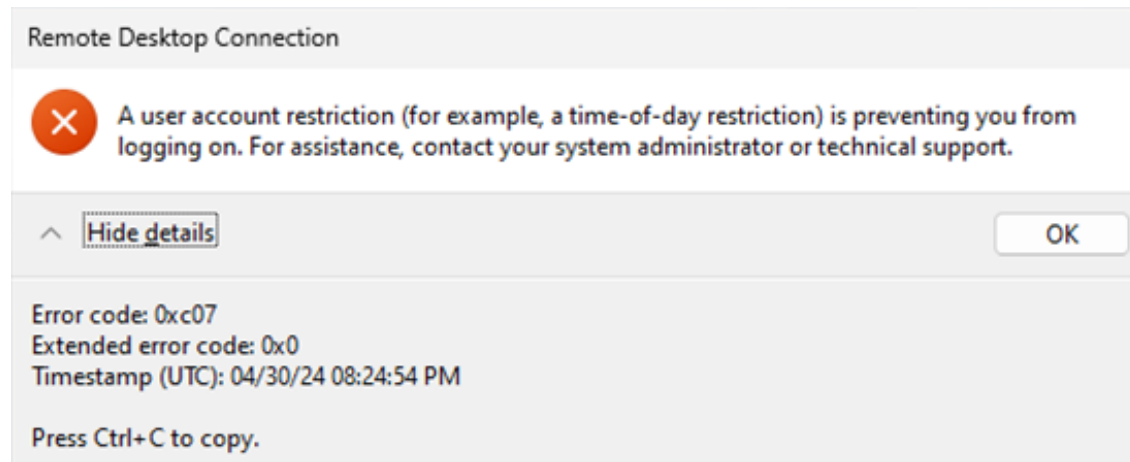
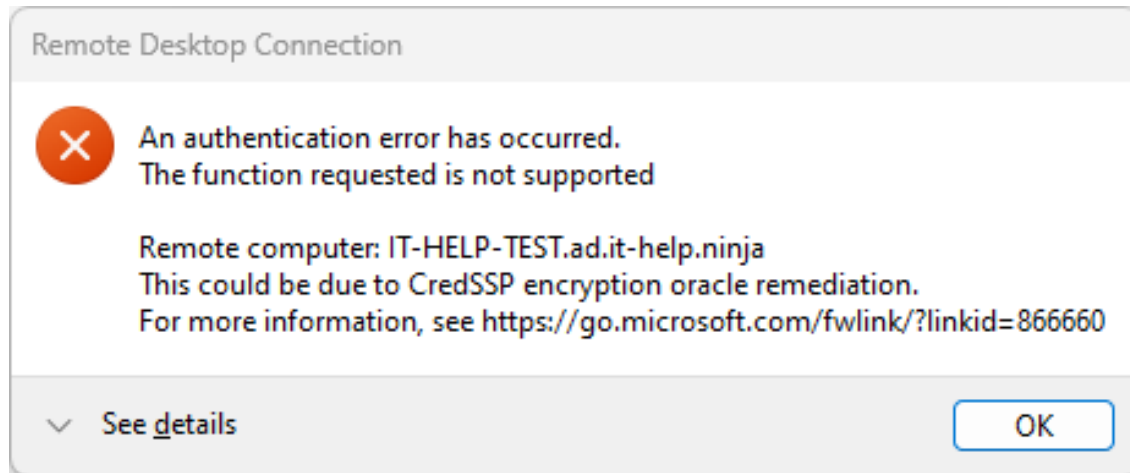
The screenshot displays the Local Group Policy Editor window. The left pane shows the tree structure with 'Local Computer Policy' expanded, and 'Security Settings' > 'Local Policies' > 'Security Options' selected. The right pane shows a list of policies. The policy 'Network security: Restrict NTLM: Outgoing NTLM traffic to remote servers' is selected and highlighted in blue. A context menu is open over this policy, showing options: 'Deny all' (selected), 'Allow all', 'Audit all', and 'Deny all' (with a warning icon). The 'Security Setting' column for this policy is 'Not Defined'.

Policy	Security Setting
Network access: Restrict clients allowed to make remote calls to SAM	Not Defined
Network access: Shares that can be accessed anonymously	Not Defined
Network access: Sharing and security model for local accounts	Classic - local u
Network security: Allow Local System to use computer identity for NTLM	Not Defined
Network security: Allow LocalSystem NULL session fallback	Not Defined
Network security: Allow PKU2U authentication requests to this computer to use online identities.	Not Defined
Network security: Configure encryption types allowed for Kerberos	Not Defined
Network security: Do not store LAN Manager hash value on next password change	Enabled
Network security: Force logoff when logon hours expire	Disabled
Network security: LAN Manager authentication level	Not Defined
Network security: LDAP client signing requirements	Negotiate signi
Network security: Minimum session security for NTLM SSP based (including secure RPC) clients	Require 128-bit
Network security: Minimum session security for NTLM SSP based (including secure RPC) servers	Require 128-bit
Network security: Restrict NTLM: Add remote server exceptions for NTLM authentication	Not Defined
Network security: Restrict NTLM: Add server exceptions in this domain	Not Defined
Network security: Restrict NTLM: Audit Incoming NTLM Traffic	Not Defined
Network security: Restrict NTLM: Audit NTLM authentication in this domain	Not Defined
Network security: Restrict NTLM: Incoming NTLM traffic	Deny all domain
Network security: Restrict NTLM: NTLM authentication in this domain	Not Defined
Network security: Restrict NTLM: Outgoing NTLM traffic to remote servers	Not Defined

Désactiver l'authentification NTLM entrante



Messages d'erreur avec NTLM désactivé




Configuration DNS

- Diagnostics de résolution DNS
- Correction de la résolution DNS
- Résolution NetBIOS et LLMNR

Diagnostiquer les problèmes de DNS

Pour diagnostiquer les problèmes de DNS, voici deux façons de faire :

- [NirSoft DNSQuerySniffer](#) 
 - Meilleur outil – intercepte les appels d'API DnsClient Windows
 - Fonctionne même avec les résultats DNS en cache client local
- Wireshark avec filtre « dns »
 - Attention – les résultats de requêtes DNS peuvent être en cache local
 - Il n'y a donc pas de requête DNS visible pour chaque résolution de nom
 - Utilisez en dernier recours si vous n'avez pas de meilleur outil de diagnostic

DNSQuerySniffer - Ethernet, 10.10.0.230, Microsoft Hyper-V Network Adapter

File Edit View Options Help

Host Name	Port Number	Query ID	Request Type	Request Time	Response Time	Duration	Response Code	Records Count	A	CNAME
IT-HELP-TEST.ad.it-help.ninja	49804	3D70	A	5/1/2024 11:28:17 AM.558	5/1/2024 11:28:17 AM.584	25.720 ms	Name Error	1		
browser.events.data.msn.com	55211	49CC	A	5/1/2024 11:28:18 AM.244	5/1/2024 11:28:18 AM.248	4.138 ms	Ok	3	52.168.117.170	global.asimov.e
browser.events.data.msn.com	55192	C4CC	A	5/1/2024 11:28:18 AM.244	5/1/2024 11:28:18 AM.249	5.151 ms	Ok	3		global.asimov.e
ctldl.windowsupdate.com	49804	22F3	A	5/1/2024 11:28:50 AM.744	5/1/2024 11:28:50 AM.749	5.072 ms	Ok	4	100.233.310.173, 100.233.	
arc-ring.msedge.net	49804	1AC5	A	5/1/2024 11:29:19 AM.617	5/1/2024 11:29:19 AM.632	14.561 ms	Ok			
mcr-ring.msedge.net	49804	B6DC	A	5/1/2024 11:29:19 AM.696	5/1/2024 11:29:19 AM.701	5.132 ms	Na			
IT-HELP-TEST.ad.it-help.ninja	49804	083E	A	5/1/2024 11:29:25 AM.540	5/1/2024 11:29:25 AM.541	1.008 ms	Na			
ctldl.windowsupdate.com	49804	0BEA	A	5/1/2024 11:29:50 AM.766	5/1/2024 11:29:50 AM.771	4.671 ms	Ok			

8 item(s), 1 Selected NirSoft Freeware. <https://www.nirsoft.net>

Remote Desktop Connection

Computer: IT-HELP-TEST.ad.it-help.ninja

User name: None specified

You will be asked for credentials when you connect.

Show Options Connect

Remote Desktop Connection

Remote Desktop can't find the computer "IT-HELP-TEST.ad.it-help.ninja". This might mean that "IT-HELP-TEST.ad.it-help.ninja" does not belong to the specified network. Verify the computer name and domain that you are trying to connect to.

See details OK

Properties

Host Name: IT-HELP-TEST.ad.it-help.ninja

Port Number: 49804

Query ID: 3D70

Request Type: A

Request Time: 5/1/2024 11:28:17 AM.558

Response Time: 5/1/2024 11:28:17 AM.584

Duration: 25.720 ms

Response Code: Name Error

Records Count: 1

A:

CNAME:

AAAA:

NS:

MX:

SOA: Admin: azuredns-hostmaster.microsoft.com, Primary Ser

PTR:

SRV:

TEXT:

Source Address: 10.10.0.230

Destination Address: 10.10.0.2

IP Country:

OK

Corriger la résolution de DNS (1/2)

Solution #1 : Changer les serveurs DNS du client

- Problématique quand on ne veut pas devenir dépendant du DNS du domaine
- Ne pas faire l'erreur d'utiliser 8.8.8.8 ou 1.1.1.1 + le serveur DNS du domaine
 - Le client DNS va utiliser la réponse de l'un sans essayer l'autre 😊
 - Le symptôme est assez flagrant : erreurs de résolution aléatoires

Solution #2 : Règle NRPT du client DNS Windows

- `Add-DnsClientNrptRule -Namespace ".ad.it-help.ninja" -NameServers @('10.10.0.3')`
- « `Get-DnsClientNrptRule` | `Remove-DnsClientNrptRule -Force` » pour l'enlever
- Affecte la plupart des applications Windows utilisant le DnsClient API
- L'outil `nslookup` n'est pas affecté puisqu'il utilise son propre stack DNS

Solution #3 : Règle de *Conditional Forwarding* dans le serveur DNS existant

- Redirige les requêtes pour un suffixe DNS vers un serveur DNS spécifié

Corriger la résolution de DNS (2/2)

Solution #4 : Modifier le fichier « hosts » sur le client temporairement

- Solution rapide et efficace, mais difficile à gérer sur plusieurs postes
- Fonctionne uniquement pour la résolution de nom (A ou AAAA records)

Solution #5 : Utiliser MsRdpEx avec UserSpecifiedServerName

- Installez MsRdpEx (<https://github.com/Devolutions/MsRdpEx>)
- Ajoutez UserSpecifiedServerName:s:<nom> dans le fichier .RDP
- Connectez-vous avec l'adresse IP : le nom spécifié sera utilisé
- Le nom spécifié sera aussi utilisé pour la validation TLS et Kerberos.

Résolution de NetBIOS et LLMNR

Pourquoi est-il possible de résoudre le nom de machine, mais pas le FQDN?

- Le *network discovery* avec NetBIOS et LLMNR s'en est occupé pour vous!
- Par mesure de sécurité, ces protocoles devraient être désactivés et remplacés par DNS
- Si le FQDN ne fonctionne pas, c'est que vous avez un problème de configuration DNS client

En résumé : utilisez toujours le Fully-Qualified Domain Name (FQDN) pour établir la connexion.
Si ça ne fonctionne pas, vous avez un problème de configuration DNS à corriger.

L'utilisation du FQDN est importante pour Kerberos de toute façon!

Bon nom: IT-HELP-TEST.ad.it-help.ninja (DNS)

Mauvais nom: IT-HELP-TEST (NetBIOS ou LLMNR, la plupart du temps)

Détection du KDC

- Commande nltest /dsgetdc
- Commande Resolve-DnsName
- Commande nslookup

Commande nltest /dsgetdc

Pour tester si Windows est capable de trouver le contrôleur de domaine, utilisez la commande **nltest /dsgetdc:<nom du domaine>**

```
nltest /dsgetdc:ad.it-help.ninja
      DC: \\IT-HELP-DC.ad.it-help.ninja
      Address: \\10.10.0.3
      Dom Guid: 4a68d84b-2f13-45bd-a519-10b58299f204
      Dom Name: ad.it-help.ninja
      Forest Name: ad.it-help.ninja
      Dc Site Name: Default-First-Site-Name
      Our Site Name: Default-First-Site-Name
      Flags: PDC GC DS LDAP KDC TIMESERV WRITABLE DNS_DC DNS_DOMAIN
      DNS_FOREST CLOSE_SITE FULL_SECRET WS DS_8 DS_9 DS_10 KEYLIST
      The command completed successfully
```


Commande Resolve-DnsName (1/2)

Pour tester la détection du contrôleur de domaine par DNS SRV record :

```
Resolve-DnsName -Type SRV _ldap._tcp.dc._msdcs.<nom du domaine>
```

```
Resolve-DnsName -Type SRV _kerberos._tcp.dc._msdcs.<nom du domaine>
```

La commande accepte aussi un serveur DNS explicite avec le paramètre -Server :

```
Resolve-DnsName -Type SRV _ldap._tcp.dc._msdcs.ad.it-help.ninja | Format-List
```

```
Name       : _ldap._tcp.dc._msdcs.ad.it-help.ninja
Type        : SRV
TTL         : 600
NameTarget  : it-help-dc.ad.it-help.ninja
Priority    : 0
Weight      : 100
Port        : 389
```

```
Name       : it-help-dc.ad.it-help.ninja
QueryType   : A
TTL         : 3600
Section     : Additional
IP4Address  : 10.10.0.3
```

Commande Resolve-DnsName (2/2)

La liste des contrôleurs de domaine est aussi disponible avec les *DNS A records* directement sur le nom de domaine DNS (ad.it-help.ninja), mais pas le nom NetBIOS (IT-HELP)

```
Resolve-DnsName ad.it-help.ninja -Type A | Format-List
```

```
Name       : ad.it-help.ninja
Type        : A
TTL         : 600
DataLength  : 4
Section     : Answer
IPAddress   : 10.10.0.3
```

```
Resolve-DnsName IT-HELP -Type A | Format-List
```

```
Resolve-DnsName : IT-HELP : The filename, directory name, or volume label syntax is incorrect
At line:1 char:1
```

```
+ Resolve-DnsName IT-HELP -Type A | Format-List
```

```
+ ~~~~~
```

```
    + CategoryInfo          : ResourceUnavailable: (IT-HELP:String) [Resolve-DnsName],
Win32Exception
    + FullyQualifiedErrorId : ERROR_INVALID_NAME,Microsoft.DnsClient.Commands.ResolveDnsName
```

Commande nslookup

Pour tester la détection du contrôleur de domaine par DNS SRV record :

```
nslookup -type=srv _ldap._tcp.dc._msdcs.<nom du domaine>  
nslookup -type=srv _kerberos._tcp.dc._msdcs.<nom du domaine>
```

La commande accepte aussi un serveur DNS explicite comme dernier paramètre optionnel :

```
nslookup -type=srv _kerberos._tcp.dc._msdcs.ad.it-help.ninja  
Server: UnKnown  
Address: 10.10.0.3
```

```
_kerberos._tcp.dc._msdcs.ad.it-help.ninja      SRV service location:  
    priority      = 0  
    weight        = 100  
    port          = 88  
    svr hostname  = it-help-dc.ad.it-help.ninja  
it-help-dc.ad.it-help.ninja    internet address = 10.10.0.3
```

Kerberos sur macOS

- Corriger la résolution de DNS sur macOS
- Détection automatique du KDC sur macOS
- Configuration de krb5.conf sur macOS

Corriger la résolution de DNS sur macOS (1/2)

La solution la plus simple est d'utiliser les serveurs DNS du domaine sur macOS, mais il existe une solution simple équivalente aux règles NRPT de Windows pour de la résolution conditionnelle :

```
sudo mkdir -p /etc/resolver
sudo bash -c 'cat > /etc/resolver/ad.it-help.ninja <<EOF
domain ad.it-help.ninja
search ad.it-help.ninja
nameserver 10.10.0.3
EOF'
```

À la suite de cette modification, les requêtes DNS finissant par « ad.it-help.ninja » utiliseront le serveur DNS du domaine (10.10.0.3, ou IT-HELP-DC.ad.it-help.ninja dans mon environnement de laboratoire).

Corriger la résolution DNS sur macOS (2/2)

Si l'utilisation du DNS n'est pas possible (par exemple, avec un tunnel SSH ou la redirection de port), ajoutez des entrées au fichier **/etc/hosts** manuellement pour permettre l'utilisation du FQDN de machine :

```
10.10.0.3      IT-HELP-DC.ad.it-help.ninja
10.10.0.10     IT-HELP-TEST.ad.it-help.ninja
```

Cependant, la détection de KDC automatique par DNS SRV ne sera pas possible sans DNS complet.

Détection automatique du KDC sur macOS

Si la détection par DNS SRV est possible (machine sur le bon réseau, avec le bon DNS) alors aucune configuration supplémentaire n'est nécessaire : le KDC sera trouvé automatiquement.

```
nslookup -type=srv _kerberos._tcp.dc._msdcs.ad.it-help.ninja
```

```
Server:          10.10.0.3
```

```
Address:         10.10.0.3#53
```

```
_kerberos._tcp.dc._msdcs.ad.it-help.ninja      service = 0 100 88 it-help-dc.ad.it-help.ninja.
```

Pour confirmer, utilisez la commande kinit pour obtenir un jeton Kerberos:

```
kinit ProtectedUser@ad.it-help.ninja
```

```
ProtectedUser@ad.it-help.ninja's password:***
```

```
klist
```

```
Credentials cache: API:CAA29FD0-DE6E-40AF-8BD0-AD68D31522C2
```

```
Principal: ProtectedUser@AD.IT-HELP.NINJA
```

Issued	Expires	Principal
May 1 17:46:11 2024	May 1 21:46:11 2024	krbtgt/ad.it-help.ninja@AD.IT-HELP.NINJA

Configuration de krb5.conf sur macOS

Si la détection par DNS SRV n'est pas possible, il faut configurer manuellement **/etc/krb5.conf** :

```
[libdefaults]
    dns_lookup_kdc = false
    dns_lookup_realm = false

[realms]
    ad.it-help.ninja = {
        kdc = tcp/IT-HELP-DC.ad.it-help.ninja
    }
    AD.IT-HELP.NINJA = {
        kdc = tcp/IT-HELP-DC.ad.it-help.ninja
    }

[domain_realm]
    .ad.it-help.ninja = ad.it-help.ninja
```

La section **[realms]** doit contenir le Kerberos *realm* en majuscules et minuscules pour éviter les problèmes liés à la sensibilité de casse du client Kerberos de macOS. (À l'interne, le realm est souvent en majuscules.)

Carte à puce virtuelle Windows

- Contraintes de la carte à puce virtuelle
- Création de la carte à puce virtuelle
- Importation de certificate dans la carte à puce
- Utilisation de la carte à puce dans mstsc
- Redirection des cartes à puce
- Gestion des cartes à puce virtuelles

Contraintes de la carte à puce virtuelle

La carte à puce virtuelle Windows est pratique pour faire des tests, avec certaines limitations :

- Il faut un *Trusted Platform Module* (TPM).
 - Avec Hyper-V, il faut une VM Gen2 avec vTPM
- Dans une session RDP, Windows ne voit que les cartes à puce redirigées du client.
- Pour tester dans une VM, il faut une solution utilisant la session « console » (non virtuelle)
 - Accès Hyper-V en *Basic Mode* sans le *Enhanced Session Mode*
 - [NICE DCV](#) rend l'utilisation de la session console plus facile que RDP
- Une machine physique Windows 11 avec TPM est probablement plus simple qu'une VM
- La gestion des cartes à puce virtuelles Windows nécessite des droits administrateurs locaux

Création de la carte à puce virtuelle

Création de la carte à puce virtuelle avec un test de PIN par défaut (12345678) :

```
tpmvscmgr.exe create /name TestVSC /pin default /adminkey random /generate
```

Using default PIN: 12345678

Creating TPM Smart Card...

Initializing the Virtual Smart Card component...

Creating the Virtual Smart Card component...

Initializing the Virtual Smart Card Simulator...

Creating the Virtual Smart Card Simulator...

Initializing the Virtual Smart Card Reader...

Creating the Virtual Smart Card Reader...

Waiting for TPM Smart Card Device...

Authenticating to the TPM Smart Card...

Generating filesystem on the TPM Smart Card...

TPM Smart Card created.

Smart Card Reader Device Instance ID = ROOT\SMARTCARDREADER\0000

Importation de certificat dans la carte à puce

Importation du certificat client dans la carte à puce virtuelle :

```
$PIN = "12345678"  
$PfxFile = ".\ProtectedUser.pfx"  
$PfxPass = "cert123!"  
$CSP = "Microsoft Base Smart Card Crypto Provider"  
certutil.exe -csp $CSP -p $PfxPass -pin $PIN -f -importPFX $PfxFile
```

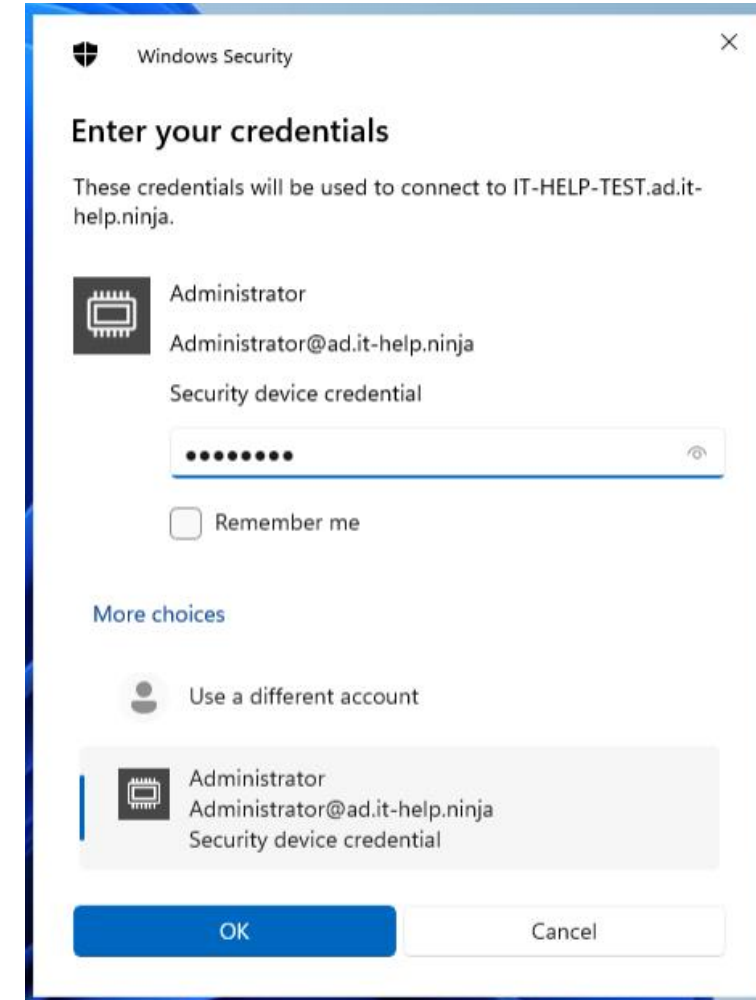
Certificate "ProtectedUser" added to store.

CertUtil: -importPFX command completed successfully.

Utilisation de la carte à puce dans mstsc

Lors de la connexion, cliquez sur **More choices** : le certificat client devrait maintenant faire partie de la liste. Entrez le PIN configuré précédemment lors de la création de la carte à puce virtuelle Windows.

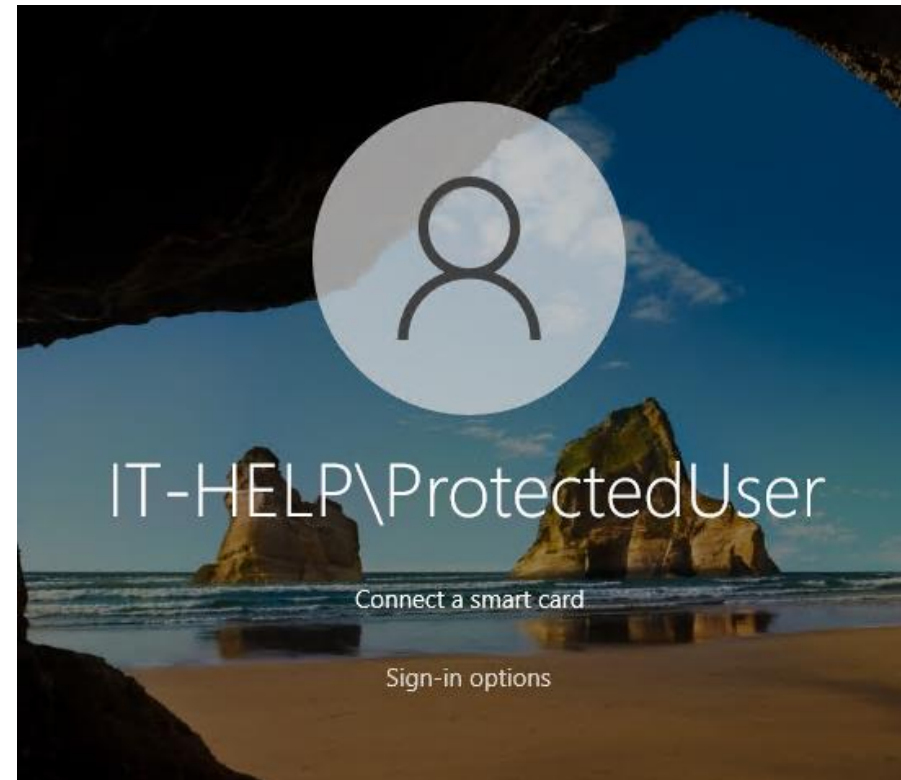
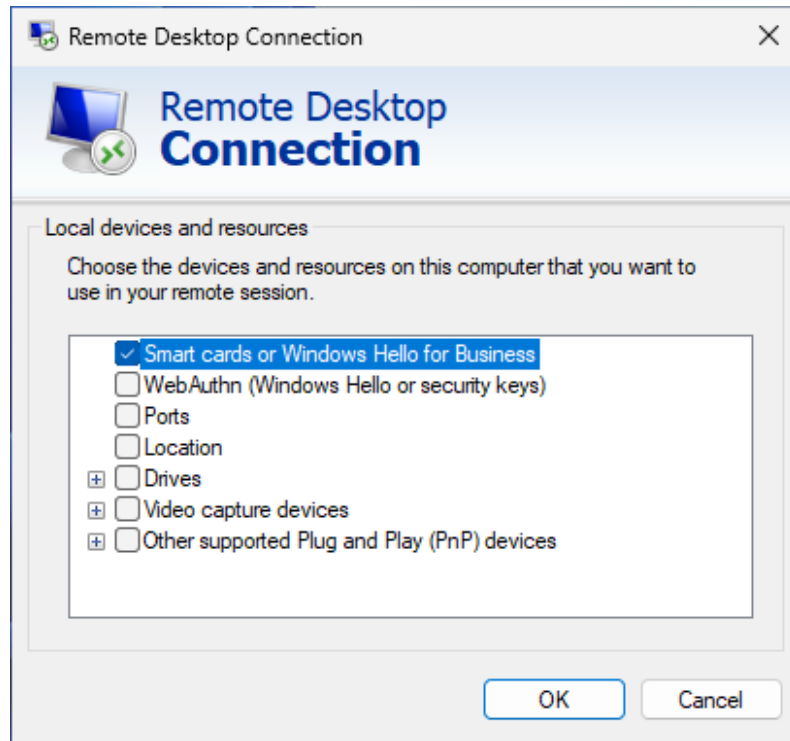
Si ça ne fonctionne pas, pas de panique! J'ai écrit un blogue sur le sujet: <https://awakecoding.com/posts/rdp-smartcard-logout-user-name-does-not-exist>



Redirection des cartes à puce

La redirection des cartes à puce est nécessaire pour compléter l'authentification.

- Si Winlogon affiche *Connect a smartcard*, la **redirection des cartes à puce n'est pas activée**.
- Vérifiez que le fichier .RDP contient la ligne `redirectsmartcards:i:1`



Gestion des cartes à puce virtuelles

Pour avoir la liste des cartes à puce virtuelles :

```
Get-PnpDevice "ROOT\SMARTCARDREADER\*" | Select-Object Name, InstanceId
```

Name	InstanceId
----	-----
TestVSC	ROOT\SMARTCARDREADER\0000

Pour supprimer une carte à puce virtuelle :

```
tpmvscmgr.exe destroy /instance ROOT\SMARTCARDREADER\0000
```

```
Destroying TPM Smart Card...
Initializing the Virtual Smart Card Reader...
Destroying the Virtual Smart Card Reader...
Initializing the Virtual Smart Card Simulator...
Destroying the Virtual Smart Card Simulator...
Initializing the Virtual Smart Card component...
Destroying the Virtual Smart Card component...
TPM Smart Card destroyed.
```

Gestion des certificats multiples

Pour avoir la liste des certificats sur une carte à puce virtuelle (très verbose) :

```
certutil -scinfo -silent
```

```
===== Certificate 0 =====  
--- Reader: Microsoft Virtual Smart Card 0  
--- Card: Identity Device (Microsoft Generic Profile)  
Provider = Microsoft Smart Card Key Storage Provider  
Key Container = ProtectedUser-5b62c36c-6b1c-4689--64376
```

Pour supprimer un certificat d'une carte à puce, il faut utiliser le **key container** :

```
$CSP = "Microsoft Base Smart Card Crypto Provider"  
$KeyContainer = "ProtectedUser-5b62c36c-6b1c-4689--64376"  
certutil -csp $CSP -delkey $KeyContainer
```


Kerberos KDC Proxy

- Qu'est-ce qu'un KDC proxy
- Service WPAD et KDC proxy
- TGT de machine et KDC proxy
- Configuration globale du KDC proxy
- Révocation de certificats et KDC proxy
- Formats KDCProxyName et KDCProxyURL
- Option de fichier RDP KDCProxyName
- Option de fichier RDP KDCProxyURL

Qu'est-ce qu'un KDC proxy

Un [KDC proxy](#) est un service HTTPS servant à relayer les messages Kerberos vers le KDC.

- Protocole simple défini dans [\[MS-KKDCP\]](#)
- HTTP POST sur /KdcProxy avec le message Kerberos dans le contenu de la requête
- C'est tout! Un KDC proxy donne un KDC *line-of-sight* nécessaire à Kerberos

La partie simple s'arrête ici : toutes les difficultés en lien avec le KDC proxy résident dans la configuration et l'injection des paramètres dans Windows ou dans le client RDP

...et dans les multiples limitations et contraintes non documentées de l'implémentation de Microsoft, sans compter les bogues non corrigés et les nombreux pièges.

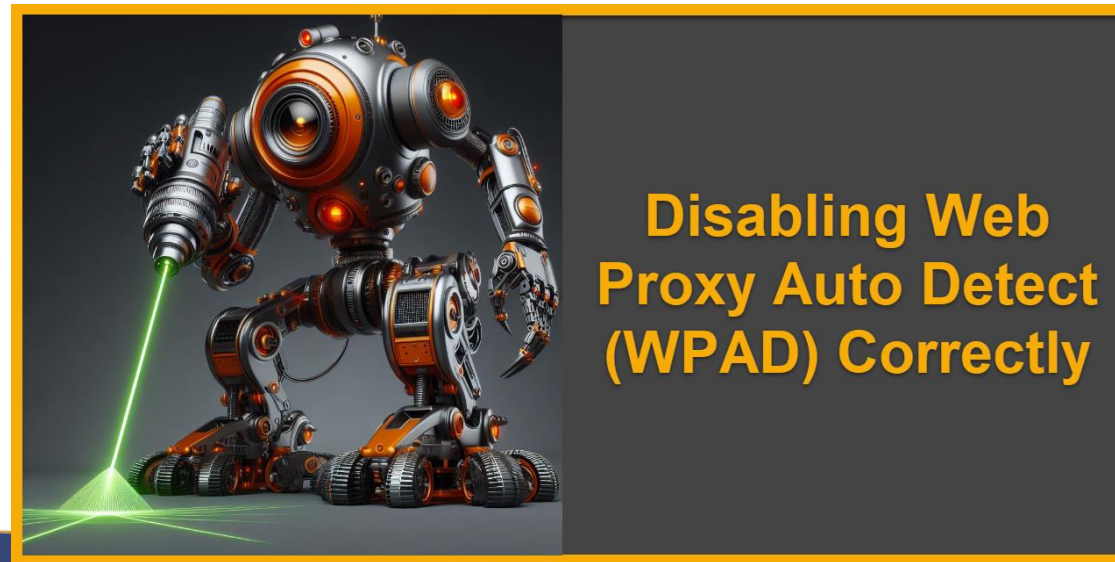
Service WPAD et KDC proxy

Désactiver le service Web Proxy Auto Detect (WPAD) brise le KDC proxy.

Il faut désactiver la **fonctionnalité** WPAD, et laisser le **service** WPAD activé:

```
New-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\WinHttp' -  
Name DisableWpad -Value 1 -Force
```

<https://awakecoding.com/posts/disabling-web-proxy-auto-detect-wpad-correctly/>



TGT de Machine et KDC proxy

Le Kerberos TGT (*Ticket-Granting-Ticket*) de la machine n'est pas obtenu automatiquement au démarrage lorsqu'un KDC proxy est utilisé, ce qui brise Kerberos

Une façon de contourner le problème est d'utiliser une tâche planifiée pour lancer la commande `klist.exe get krbtgt` automatiquement au démarrage.

<https://awakecoding.com/posts/fix-kerberos-machine-tgt-fetching-on-startup/>



Configuration globale du KDC proxy (1/2)

```
$KdcServerUrl = "https://gateway.ad.it-help.ninja/KdcProxy"  
$KerberosRealm = "ad.it-help.ninja"
```

```
$KdcUrl = [Uri] $KdcServerUrl  
$KdcHost = $KdcUrl.Host  
$KdcPort = $KdcUrl.Port  
$KdcPath = $KdcUrl.AbsolutePath.TrimStart('/')  
if ([string]::IsNullOrEmpty($KdcPath)) {  
    $KdcPath = "KdcProxy"  
}  
$KdcProxyServer = "<https $KdcHost`:$KdcPort`:$KdcPath />"
```

```
$KerberosReg = "HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\Kerberos"  
New-Item "$KerberosReg\Parameters" -Force  
New-Item "$KerberosReg\KdcProxy\ProxyServers" -Force  
New-ItemProperty $KerberosReg -Name "KdcProxyServer_Enabled" -Type DWORD -Value 1 -Force  
New-ItemProperty "$KerberosReg\KdcProxy\ProxyServers" -Name $KerberosRealm -Value  
$KdcProxyServer -Force  
if ($ForceProxy) {  
    New-ItemProperty "$KerberosReg\Parameters" -Name "ForceProxy" -Type DWORD -Value 1 -Force  
}
```

Configuration globale du KDC proxy (2/2)

Les appels au KDC proxy se font à partir de lsass.exe, le service système du *Local Security Authority* (LSA) de Windows **qui ne peut pas être redémarré sans redémarrer Windows**.

La seule façon de forcer une relecture de la configuration du KDC proxy sans redémarrer est à l'aide d'un *group policy update notification* (commande `gpupdate /force`).

- Seul hic : ça ne marchera pas sans un *line-of-sight* avec le contrôleur de domaine!

Point important : sans l'option **ForceProxy**, lsass.exe tentera quand même une connexion directe avec le KDC, ce qui peut faussement laisser croire que le KDC proxy fonctionne.

Conclusion : pour appliquer un changement de configuration globale du KDC proxy, **vous devez redémarrer la machine**, ce qui s'avère pénible si ça ne marche pas du premier coup.

Révocation de certificats et KDC proxy

Le KDC proxy **utilise obligatoirement HTTPS** comme couche de transport

- Le certificat doit aussi être automatiquement validable (pas de *self-signed*).
- Avec un CA interne, faites attention au *Certificate Revocation List* (CRL)!
- Vérifiez les événements dans *Applications and Services Logs\Microsoft\Windows\CAPI2*

À moins d'exposer l'URL de distribution CRL à l'externe, vous aurez des problèmes sporadiques de vérification de l'état de révocation utilisée pour le KDC proxy.

Pour contourner le problème :

```
New-ItemProperty -Path  
"HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\Kerberos\Parameters" -Name  
UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors -Value 1 -Force
```

Pour invalider le cache CRL :

```
& certutil.exe "-urlcache" "crl" "delete"  
& certutil.exe "-setreg" "chain\ChainCacheResyncFiletime" "@now"
```

Formats KDCProxyName et KDCProxyURL

KDCProxyURL : `https://<host>[:<port>][/path]`

KDCProxyName : `<host>[:<port>][:<path>]`

Registre Windows : `<https KDCProxyName />`

Le format KDCProxyName ressemble à une URL sans en être une : la portion *path* de l'URL est séparée par un deux-points (':') au lieu d'un slash ('/'). Il est très facile de se tromper!

Le format KDCProxyName omet le protocole (`https://`) et présume HTTPS en tout temps.

Pour éviter les erreurs, utilisez un bout de code PowerShell pour faire la conversion entre KDCProxyURL et KDCProxyName, et utilisez l'URL comme paramètre à vos scripts.

MsRdpEx accepte KDCProxyURL directement comme option de fichier .RDP 😊

Option de fichier RDP KDCProxyName

L'option de [fichier RDP documentée *KDCProxyName*](#) sert à injecter un KDC proxy pour une connexion RDP sans devoir configurer le KDC proxy globalement sur la machine client.

Ce que la documentation ne dit pas :

- **L'option KDCProxyName est ignorée pour les connexions n'utilisant pas RD Gateway ou Azure Virtual Desktop, donc elle ne fonctionne pas en RDP « normal ».**
- La longueur maximale de KDCProxyName est de 255 caractères.
- Le format KDCProxyName complet avec port et *path* explicite n'est pas clarifié.

Depuis 2021, j'essaie de faire corriger les problèmes ci-dessus par Microsoft, en vain.

Option de fichier RDP KDCProxyURL

L'option de fichier [KDCProxyURL de MsRdpEx](#) vient combler les lacunes de KDCProxyName :

- Accepte le format d'URL standard au lieu du format étrange de KDCProxyName
- Fonctionne en tout temps – pas seulement avec RD Gateway ou Azure Virtual Desktop
- Accepte des URL de KDC proxy de longueur variable (pas de limite de 255 caractères)

L'option KDCProxyURL est utilisée avec [Remote Desktop Manager et Devolutions Gateway](#) pour faire du *just-in-time* RDP + KDC *proxying* avec des jetons temporaires.

Vous pouvez aussi déployer MsRdpEx séparément juste pour bénéficier de l'option KDCProxyURL sans les limitations de KDCProxyName.

Remote Desktop Gateway

- Introduction RD Gateway
- KDC Proxy avec RD Gateway

Introduction RD Gateway

Remote Desktop Gateway (anciennement TS Gateway) est une sorte de VPN spécifique à RDP :

- Le client RDP s'authentifie avec RD Gateway pour demander l'ouverture d'un tunnel de connexion.
- La *Connection Authorization Policy* (CAP) limite quelles machines ou utilisateurs peuvent se connecter.
- La *Resource Authorization Policy* (RAP) limite quelles machines (ressources) peuvent être accédées.

Références :

- [Deploy the Remote Desktop Gateway role](#)
- [Remote Desktop Services – Access from anywhere](#)
- [Remote Desktop Services architecture](#)

KDC Proxy avec RD Gateway

Le [KDC proxy](#) est la seule façon de faire fonctionner l'authentification Kerberos avec RD Gateway.

Voici comment activer le service *KDC proxy* sur la route /KdcProxy du RD Gateway :

```
& netsh http add urlacl "url=https://+:443/KdcProxy" 'user="NT AUTHORITY\Network Service"'

$KdcProxyReg = "HKLM:\SYSTEM\CurrentControlSet\Services\KPSSVC\Settings"
New-ItemProperty $KdcProxyReg "HttpsClientAuth" -Type DWORD -Value 0 -Force
New-ItemProperty $KdcProxyReg "DisallowUnprotectedPasswordAuth" -Type DWORD -Value 0 -Force
New-ItemProperty $KdcProxyReg "HttpsUrlGroup" -Type MultiString -Value "+`:443" -Force

Set-Service -Name KPSSVC -StartupType Automatic
Start-Service -Name KPSSVC
```

Les options RDGIsKDCProxy et KDCProxyName du fichier .RDP sont seulement utilisées pour Kerberos dans la connexion RDP, mais pas pour l'ouverture du tunnel avec RD Gateway.

Si vous avez des questions, je suis
disponible tout au long de ITSec.

Venez me voir 😊

<https://twitter.com/awakecoding>
mamoreau@devolutions.net
<https://awakecoding.com/>

Merci!

ITSec

FORMATION