Total Network Inventory

Scan Account Security Guide

Least-Privilege Scanning — Domain & Local Admin Group Setup + gMSA

Version 2.0 — April 2026

Internal IT Security Use Only

1. The Security Risk — Why Domain Admin Should Not Be Used

When a Domain Administrator account is stored in TNI for periodic scanning, it creates the following chain of risk:

- TNI stores scan credentials in the PostgreSQL database, encrypted with AES-256.
- Any user who has TNI installed can access the PostgreSQL connection parameters through the application's configuration settings.
- Even a read-only database user, once connected to PostgreSQL, can query the tables where encrypted credentials are stored.
- A Domain Administrator password — even AES-256 encrypted — sitting in an accessible database is a high-value target. If the encryption key is ever compromised, or if the attacker bypasses the application layer, full domain control is at risk.

> Security Risk: Using a Domain Administrator account for routine TNI scanning violates the principle of least privilege. A compromised scan credential would give an attacker unrestricted access to every machine and system in your Active Directory domain.

> Best Practice: Create a dedicated domain service account with only local administrator rights on the target computers. TNI only needs local admin rights on the scanned machine — not domain-wide elevated privileges.

2. The Solution — Dedicated Scan Service Account

2.1 What the Account Needs

The TNI scan account requires exactly one permission on each target computer:

Requirement	Why it is needed	Scope
Member of local Administrators group on each scanned computer	TNI uses WMI, RPC, and SMB to read hardware, software, and OS data. Windows blocks these calls for non-local-admins by default.	Local to each target machine only
Standard domain user account (no Domain Admin)	Authentication to join the domain and be distributed via Group Policy. No elevated domain privileges required.	Domain — authentication only
Firewall exception for WMI / Remote Admin	TNI uses WMI (TCP 135 + dynamic ports) and SMB (TCP 445). These must be allowed from the TNI server.	Network / GPO firewall rules

> Security Risk: Do NOT add the scan account to the Domain Admins group. This defeats the purpose. Local admin rights on individual machines are sufficient.

2.2 Account Naming Recommendation

Use a clear, identifiable name that signals the account's limited purpose:

- svc-tni-scan (preferred — service account prefix)
- tni_scanner (acceptable alternative)
- IT-Inventory-Scan (descriptive alternative)

> Best Practice: Prefix service accounts with 'svc-' so they are immediately recognisable in logs and AD as non-human accounts. Set the password to never expire but ensure it is complex (20+ characters). Document it in your password vault.

3. Step 1 — Create the Domain User Account

Perform this on a Domain Controller or any machine with Active Directory Users and Computers (ADUC) installed.

Method A — Active Directory Users and Computers (GUI)

1. Open ADUC: Press Windows + R → type dsa.msc → press Enter.
2. Navigate to the correct OU: Expand your domain → right-click the Service Accounts OU → click New → User.
3. Fill in the account details: First name: TNI / Last name: Scanner / User logon name: svc-tni-scan

4. Set the password: Use a strong password (20+ characters). Tick Password never expires. Untick User must change password at next logon.
5. Verify: Right-click the new account → Properties → confirm it is NOT a member of Domain Admins. It should only be in Domain Users.

Method B — PowerShell (faster, scriptable)

Run the following on a Domain Controller or with RSAT tools installed:

```
# Create the scan service account
New-ADUser `
  -Name "TNI Scanner" `
  -SamAccountName "svc-tni-scan" `
  -UserPrincipalName "svc-tni-scan@yourdomain.com" `
  -Path "OU=ServiceAccounts,DC=yourdomain,DC=com" `
  -AccountPassword (ConvertTo-SecureString "YourStr0ngP@ssword!" -AsPlainText -
Force) `
  -PasswordNeverExpires $true `
  -CannotChangePassword $true `
  -Enabled $true


# Verify it is NOT in any elevated group
Get-ADUser "svc-tni-scan" -Properties MemberOf | Select-Object -ExpandProperty
MemberOf
```

> Note: Replace yourdomain.com and the password with values matching your environment.

4. Step 2 — Add the Account to Local Administrators on Target Machines

There are two approaches. Choose the one that matches your environment:

	Approach A: Domain Security Group + GPO	Approach B: GPO Preferences (Item-Level Targeting)
Best for	Large environments, all computers in scope	Environments needing granular targeting per OU
Risk of removing existing admins	High if using Restricted Groups	Low — Update action only adds, does not remove
Recommended	Yes — simpler and cleaner	Yes — preferred for mixed environments

5. Approach A — Domain Security Group + Group Policy

Phase 1 — Create a Domain Security Group

- Open ADUC → right-click Service Accounts OU → New → Group
- Group name: GRP-TNI-LocalAdmins | Scope: Global | Type: Security
- Add svc-tni-scan as a member of GRP-TNI-LocalAdmins

Phase 2 — Create and Link the GPO

- Open GPMC: Windows + R → gpmc.msc
- Navigate to the OU containing workstations/servers
- Right-click OU → Create a GPO in this domain and link it here
- Name: TNI-LocalAdmin-Policy

Phase 3 — Configure the GPO

In the Group Policy Management Editor navigate to:
Computer Configuration → Preferences → Control Panel Settings → Local Users and Groups

- Right-click → New → Local Group
- Action: Update (adds members without removing existing ones)
- Group name: Administrators (built-in)
- Add member: YOURDOMAIN\GRP-TNI-LocalAdmins
- Do NOT tick "Delete all member users" or "Delete all member groups"

> Security Risk: Never tick "Delete all member users / groups" unless you are certain you want to replace ALL local admins. This can lock out existing local admin accounts.

Phase 4 — Apply and Verify

```
# Force policy update on target computers:
gpupdate /force


# Verify local Administrators group:
Get-LocalGroupMember -Group "Administrators"
```

> Best Practice: Use gpresult /r on a target computer to confirm the TNI-LocalAdmin-Policy GPO is being applied.

6. Approach B — GPO Item-Level Targeting

Use this when you need to add the scan account to local admins only on specific computers or OUs — not your entire domain.

- Follow Phase 1-2 of Approach A, then open the GPO for editing
- Create the Local Group entry (Action: Update, Administrators, add GRP-TNI-LocalAdmins)
- Go to the Common tab → tick Item-level targeting → click Targeting
- Choose filter: Organizational Unit, Computer Name, Security Group, or WMI Query

- Combine multiple conditions using AND/OR logic

> Best Practice: Item-level targeting is the most flexible approach. You can combine conditions (e.g. OU = Workstations AND OS = Windows 11) for precise control.

7. Step 3 — Configure the Scan Account in TNI

6. Open TNI → Logins (the key icon in the left panel)
7. Click + → Add a new login
8. Protocol: Windows | Username: YOURDOMAIN\svc-tni-scan | Password: your password
9. Assign this login to the scanner task or relevant asset groups
10. Run a test scan on one computer and verify it completes successfully

> Note: TNI stores this password AES-256 encrypted in PostgreSQL. Ensure the tni_readonly PostgreSQL role cannot read the logins table — or restrict pgAdmin access entirely to the DBA only.

8. Step 4 — Configure Windows Firewall via GPO

TNI needs WMI and Remote Administration firewall exceptions on target computers. Deploy via GPO:

Navigate to: Computer Configuration → Policies → Windows Settings → Security Settings → Windows Defender Firewall → Inbound Rules

- Enable: Windows Management Instrumentation (WMI-In) — TCP 135 + dynamic RPC ports
- Enable: File and Printer Sharing (SMB-In) — TCP 445
- Enable: Remote Administration — RPC remote admin access
- Scope each rule to TNI server IP only: Rule → Scope tab → Remote IP addresses → add TNI server IP

> Best Practice: Scoping WMI firewall rules to the TNI server IP is a critical extra layer. Even if svc-tni-scan is compromised, WMI access is only possible from your designated scan server.

```
# Enable WMI firewall exception (scope to TNI server IP)
Set-NetFirewallRule -DisplayGroup "Windows Management Instrumentation (WMI)" `
  -Enabled True -Profile Domain

# Enable Remote Admin exception
Set-NetFirewallRule -DisplayGroup "Remote Administration" `
  -Enabled True -Profile Domain
```

9. Advanced Option — Group Managed Service Account (gMSA)

gMSA is the most secure method for TNI scanning. It eliminates the password from PostgreSQL entirely — the password is never stored anywhere, it is computed on demand by Active Directory.

> Security Risk: The core risk with svc-tni-scan is that its AES-256 encrypted password still exists in the PostgreSQL database. If the encryption key is compromised, the password is exposed. gMSA eliminates this risk completely.

9.1 How gMSA Authentication Works

Unlike a standard account where a password string is stored and compared, gMSA uses a cryptographic computation:

- Active Directory holds a KDS Root Key — a master secret that never leaves the DC.
- When TNI-SRV needs to authenticate, it requests the current password from AD.
- AD computes the password using: HMAC-SHA256(KDS_Root_Key + account_name + current_30day_period)
- The result — a 240-character password — is handed to Windows LSASS in memory only.
- TNI never sees the password. Windows uses it automatically for all authentication.
- After 30 days the period changes, the computed password changes automatically. No manual rotation required.

> Note: The password is never stored in PostgreSQL. An attacker who compromises the TNI database sees only the account name tni-scanner$ — there is no password to steal.

9.2 Prerequisites

- Domain functional level: Windows Server 2012 R2 or higher
- At least one Domain Controller running Windows Server 2012 R2 or higher
- KDS Root Key created on the DC (one-time setup per domain)
- TNI server must be domain-joined

9.3 Step-by-Step gMSA Setup

Step 1 — Create the KDS Root Key (one-time per domain)

Run on a Domain Controller:

```
# For production — key becomes active after 10 hours:
Add-KdsRootKey -EffectiveImmediately

# For lab/testing — activate immediately:
Add-KdsRootKey -EffectiveTime (Get-Date).AddHours(-10)
```

```
# Verify:
Get-KdsRootKey
```

> Note: The KDS Root Key only needs to be created once per domain. If one already exists, skip this step.

Step 2 — Create the gMSA account

```
New-ADServiceAccount `
  -Name "tni-scanner" `
  -DNSHostName "tni-scanner.yourdomain.com" `
  -PrincipalsAllowedToRetrieveManagedPassword "TNI-SRV$"
```

> Security Risk: The PrincipalsAllowedToRetrieveManagedPassword parameter is critical. Only the machine listed here can request the gMSA password from AD. Always specify the exact computer account (TNI-SRV$), never a broad group like Domain Computers.

Step 3 — Install the gMSA on TNI-SRV

Run on the TNI-SRV machine:

```
# Install the gMSA on this machine:
Install-ADServiceAccount -Identity "tni-scanner"

# Verify the installation:
Test-ADServiceAccount -Identity "tni-scanner"
# Must return: True
```

> Note: If Test-ADServiceAccount returns False, ensure TNI-SRV$ is listed in PrincipalsAllowedToRetrieveManagedPassword and that the KDS Root Key has propagated to all DCs.

Step 4 — Add gMSA to local Administrators via GPO

Use the same GPO approach as Sections 5-6, but add tni-scanner$ instead of svc-tni-scan:

- Create domain security group: GRP-TNI-LocalAdmins (if not already created)
- Add tni-scanner$ to GRP-TNI-LocalAdmins
- GPO: Computer Configuration → Preferences → Local Users and Groups
- Action: Update | Group: Administrators (built-in) | Add: YOURDOMAIN\GRP-TNI-LocalAdmins

```
# Verify on a target computer after gpupdate /force:
Get-LocalGroupMember -Group "Administrators"
# Should show: YOURDOMAIN\tni-scanner$
```

Step 5 — Configure TNI to use the gMSA

In TNI Logins, configure the gMSA account:

- Protocol: Windows
- Username: YOURDOMAIN\tni-scanner$ (note the $ suffix — this signals gMSA to Windows)
- Password: leave blank — Windows retrieves it automatically from AD

> Best Practice: After configuring the gMSA in TNI, run a test scan on one computer. If it fails, check that Test-ADServiceAccount returns True on TNI-SRV and that the GPO has applied to the target machine.

9.4 gMSA Risks and Mitigations

gMSA significantly reduces risk but introduces some operational considerations:

Risk	Impact	Mitigation
DC compromised	KDS Root Key exposed — all gMSA passwords computable	Protect DC as highest-value asset. Monitor DC access via Event ID 4662.
TNI-SRV compromised	Attacker can request gMSA password from AD	Harden TNI-SRV: EDR, minimal software, restrict RDP, audit all logins.
DC unavailable at service start	TNI service cannot start (no password retrieval)	Deploy minimum 2 DCs. Windows caches gMSA password for 30 days.
Broad PrincipalsAllowed list	Multiple machines can request the gMSA password	Always specify exact machine account (TNI-SRV$), never a group.

10. Scanning with Domain Admin Rights Using gMSA

There are rare scenarios where Domain Admin rights are genuinely required for scanning — for example, reading certain domain-wide configurations or scanning Domain Controllers directly. In these cases, gMSA can be used to hold Domain Admin rights without storing the password anywhere.

> Security Risk: This approach should only be used when Domain Admin rights are absolutely necessary and cannot be avoided. The blast radius if TNI-SRV is compromised remains very high — Domain Admin rights on a gMSA mean full domain control is accessible from that machine.

10.1 How It Works

The gMSA account (tni-scanner$) is added to the Domain Admins group. The password is never stored — it is computed by AD on demand. This eliminates the PostgreSQL credential risk while preserving Domain Admin capability.

- Password never appears in PostgreSQL — only the account name tni-scanner$ is stored
- Password is computed by AD, handed to Windows in memory, discarded after use
- Domain Admin rights are active — TNI-SRV can scan any machine including DCs
- If TNI-SRV is compromised, the attacker can request Domain Admin credentials from AD

10.2 Setup

```
# 1. Create gMSA (Steps 1-3 from Section 9.3 apply identically)
New-ADServiceAccount `
  -Name "tni-scanner" `
  -DNSHostName "tni-scanner.yourdomain.com" `
  -PrincipalsAllowedToRetrieveManagedPassword "TNI-SRV$"


# 2. Add gMSA to Domain Admins
Add-ADGroupMember -Identity "Domain Admins" -Members "tni-scanner$"


# 3. Verify membership
Get-ADGroupMember -Identity "Domain Admins" | Where-Object {$_.Name -eq "tni-
scanner"}


# 4. Install gMSA on TNI-SRV and configure TNI Logins
# (identical to Steps 3-5 in Section 9.3)
```

> Security Risk: Verify that only TNI-SRV$ is in PrincipalsAllowedToRetrieveManagedPassword. Any additional machine in this list can request Domain Admin credentials from AD.

10.3 Additional Hardening for Domain Admin gMSA

Because the stakes are higher with Domain Admin rights, apply these additional controls:

- Deny interactive logon for tni-scanner$ via GPO: Deny log on locally + Deny log on through RDS
- Firewall: WMI and SMB inbound allowed only from TNI-SRV IP (192.168.x.x) on all machines
- Enable audit logging for tni-scanner$ — alert on any authentication outside scheduled scan windows
- Restrict TNI-SRV: only designated administrators can RDP or log on locally
- Monitor Event ID 4768 (Kerberos ticket request) for tni-scanner$ on all DCs

```
# GPO — deny interactive logon for tni-scanner$
# Computer Configuration → Policies → Windows Settings →
# Security Settings → Local Policies → User Rights Assignment
# "Deny log on locally" → add YOURDOMAIN\tni-scanner$
```

```
# "Deny log on through Remote Desktop Services" → add YOURDOMAIN\tni-scanner$
```

11. Security Comparison — All Three Methods

Use this table to select the appropriate method based on your environment and risk tolerance:

Factor	Domain Admin (plain)	svc-tni-scan (local admin)	gMSA tni-scanner$
Password in PostgreSQL	YES — Domain Admin password	YES — service account password	NO — password never stored
Blast radius if DB compromised	Full domain control	Local admin on scanned PCs only	Nothing to steal
Can modify Active Directory	YES	NO	Only if added to Domain Admins
Can log into Domain Controllers	YES	NO	Only if Domain Admins member
Password rotation	Manual	Manual	Automatic every 30 days
Interactive login possible	YES	YES (can be denied via GPO)	NO — by design
Requires live DC to start	NO	NO	YES (cached for 30 days)
Setup complexity	Low	Low	Medium
Compliance / least privilege	FAILS	PASSES	PASSES (best)
Recommended	Never	Yes — standard environments	Yes — high-security / CII

12. Implementation Checklist

Option A — svc-tni-scan (Local Admin)

- Domain user svc-tni-scan created in AD — NOT in Domain Admins or any elevated group
- Strong password set (20+ characters), stored in password vault, password never expires
- Domain security group GRP-TNI-LocalAdmins created
- svc-tni-scan added to GRP-TNI-LocalAdmins
- GPO TNI-LocalAdmin-Policy created and linked to the correct OU(s)
- GPO uses Update action — does NOT delete existing local admins
- GPO verified with gpresult /r on at least one target computer

- net localgroup Administrators confirms GRP-TNI-LocalAdmins is present on target
- WMI and Remote Admin firewall rules enabled and scoped to TNI server IP
- TNI Logins configured with YOURDOMAIN\svc-tni-scan
- Test scan completed successfully on at least one computer
- PostgreSQL tni_readonly role restricted from accessing the logins/credentials table

Option B — gMSA tni-scanner$

- KDS Root Key created on DC (Get-KdsRootKey returns a valid entry)
- gMSA account tni-scanner$ created with PrincipalsAllowed = TNI-SRV$ only
- gMSA installed on TNI-SRV: Install-ADServiceAccount succeeded
- Test-ADServiceAccount returns True on TNI-SRV
- tni-scanner$ added to GRP-TNI-LocalAdmins (or Domain Admins if required)
- GPO adds GRP-TNI-LocalAdmins to local Administrators on target machines
- TNI Logins configured: username = YOURDOMAIN\tni-scanner$, password blank
- Test scan completed successfully on at least one computer
- Deny interactive logon GPO applied for tni-scanner$
- WMI firewall rules scoped to TNI server IP on all targets
- Event ID 4768 alerting configured for tni-scanner$ on DCs
- Minimum 2 Domain Controllers deployed for resilience