Konta usługowe MSA i gMSA

Typ: procedura **Dotyczy: Active Directory, Windows Server 2012+** **Czas: ~30 min**

Efekt: usługa albo pula aplikacji IIS uwierzytelnia się kontem, którego hasła nikt nie zna i nie wpisuje w konfiguracji — Active Directory zmienia je samo, automatycznie, zanim ktokolwiek zdąży o nim zapomnieć.

Skróty użyte na tej stronie (6)

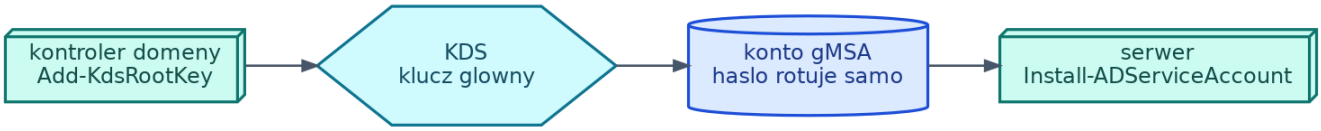
Skrót	Co znaczy
AD	Active Directory: katalog Microsoftu z kontami użytkowników, komputerów i grup w firmie
gMSA	group Managed Service Account: konto usługi, któremu Windows sam zmienia hasło
GPO	Group Policy Object: zestaw ustawień wypychanych z Active Directory na komputery
GUI	Graphical User Interface: okienka i myszka zamiast komend
IIS	Internet Information Services: serwer WWW wbudowany w Windows Server
MSA	Managed Service Account: konto usługi z automatyczną zmianą hasła, dla jednego serwera

Co realnie rozwiązuje konto usługi zarządzane

> **Pojęcie: konto usługi zarządzane (Managed Service Account).** Konto w Active Directory przeznaczone wyłącznie do uruchamiania usług i procesów, nie do logowania interaktywnego, którego hasło Active Directory zmienia samo, cyklicznie i automatycznie — bez udziału administratora i bez zapisywania go nigdzie w plikach konfiguracyjnych.

Problem, który to rozwiązuje, jest znany każdemu, kto kiedykolwiek zmieniał hasło do konta usługowego ręcznie: hasło trzeba zmienić w AD, a potem we wszystkich miejscach, gdzie jest wpisane — w konfiguracji usługi, w connection stringu, czasem w kilku aplikacjach naraz — i przy którymkolwiek z tych kroków pominiętym usługa przestaje działać, zwykle w najmniej dogodnym momencie. Konto usługi zarządzane eliminuje ten problem u źródła: hasła nigdzie nie wpisujemy, bo usługa i tak go nie potrzebuje — pobiera je automatycznie z AD w momencie startu.

Różnica MSA vs gMSA. Standardowe MSA (`Standalone Managed Service Account`) jest przypisane do jednego konkretnego komputera i nie da się go użyć na drugim. gMSA (`group Managed Service Account`) jest przypisane do grupy zabezpieczeń zawierającej komputery, którym wolno go używać — dzięki temu to samo konto usługowe obsługuje usługę rozproszoną na kilku serwerach naraz (farma IIS, klaster) bez zakładania osobnego konta na każdą maszynę. W praktyce na nowych wdrożeniach niemal zawsze wybieramy gMSA — jest funkcjonalnym nadzbiorem MSA.



Kontroler domeny generuje klucz KDS, z niego powstaje haslo konta gMSA, serwer pobiera je sam (kliknięcie otwiera pełny rozmiar)

Zanim zaczniemy

> **Z ŻYCIA WZIĘTE**
>
> Ola prosi o konto usługowe dla nowej aplikacji zespołu, rozproszonej na dwóch serwerach naraz, żeby nie musieć ręcznie synchronizować hasła między nimi po każdej wymuszonej zmianie. Zakładamy gMSA, dodajemy obie maszyny do grupy uprawnionej do pobierania hasła i instalujemy konto na każdej z nich. Od tego dnia hasło konta rotuje samo co trzydzieści dni, a żaden administrator nigdy go nie widzi ani nie wpisuje w konfiguracji.

Prace na kontrolerze domeny wymagają konta z uprawnieniami do zarządzania AD w domenie, w której chcemy użyć kont gMSA. Sprawdzamy tryb domeny i lasu — funkcja gMSA wymaga poziomu funkcjonalnego Windows Server 2012 lub wyższego:

```
(Get-ADDomain).ForestMode
(Get-ADDomain).DomainMode
```

Przebieg

1. Dodanie głównego klucza KDS

Wykonujemy raz na cały las — jeśli klucz KDS już istnieje z wcześniejszego wdrożenia gMSA, ten krok pomijamy. Ustawiamy `-EffectiveTime` na 10 godzin wstecz, żeby klucz zaczął obowiązywać natychmiast, zamiast czekać na domyślną, znacznie dłuższą propagację między kontrolerami domeny:

```
Add-KdsRootKey -EffectiveTime (Get-Date).AddHours(-10)
```

Testujemy poprawność klucza:

```
Test-KdsRootKey -KeyId (Get-KdsRootKey).KeyId
```

2. Utworzenie konta gMSA

Częstotliwość automatycznej zmiany hasła ustawiamy tylko przy tworzeniu konta — domyślnie jest to 30 dni, a zmiana tej wartości później wymaga dodatkowych kroków, więc od razu dobieramy wartość docelową:

```
New-ADServiceAccount -Name NazwaKonta -DNSHostName NazwaKonta.NazwaDomeny -
ManagedPasswordIntervalInDays LiczbaDni
```

3. Przypisanie grupy uprawnionej do pobierania hasła konta

```
Set-ADServiceAccount -Identity "NazwaKonta" -PrincipalsAllowedToRetrieveManagedPassword
"NazwaGrupy"
```

`NazwaGrupy` to grupa zabezpieczeń zawierająca konta komputerów, którym wolno korzystać z tego gMSA — nie konta użytkowników, tylko konta komputerów serwerów, na których usługa faktycznie działa.

4. Instalacja konta na serwerze docelowym

Na serwerze, który ma korzystać z konta, instalujemy przystawki PowerShell do zarządzania AD, jeśli nie są jeszcze zainstalowane:

```
Add-WindowsFeature RSAT-AD-PowerShell
```

Resetujemy bilet uwierzytelniania Kerberos, żeby serwer na pewno odświeżył swoje członkostwo w grupie z poprzedniego kroku:

```
klist purge -li 0x3e7
```

Instalujemy konto na serwerze:

```
Install-ADServiceAccount -Identity "NazwaKonta"
```

5. Przypisanie konta do usługi

Przez GUI (*Właściwości usługi* → zakładka *Logowanie* → *To konto*, wpisując `NazwaDomeny\NazwaKonta$` bez podawania hasła), albo przez PowerShell:

```
sc.exe config NazwaUslugi obj= "NazwaDomeny\NazwaKonta$" password= ""
```

Dla puli aplikacji IIS ustawiamy identyfikator tożsamości w Menedżerze IIS (*Application Pools* → *Advanced Settings* → *Identity* → *Custom account*), podając `NazwaDomeny\NazwaKonta$` również bez hasła — IIS pobiera je automatycznie z AD tak samo jak każda inna usługa systemowa.

6. Uprawnienie „Log on as a service"

Konto gMSA przypisane do usługi przez Menedżera usług albo IIS otrzymuje to uprawnienie automatycznie w większości scenariuszy. Jeśli usługa mimo poprawnej instalacji konta odmawia startu z błędem logowania, sprawdzamy wprost w polityce lokalnej (`secpol.msc` → *Local Policies* → *User Rights Assignment* → *Log on as a service*) albo w odpowiednim GPO, czy `NazwaDomeny\NazwaKonta$` rzeczywiście tam figuruje.

Weryfikacja

```
Test-ADServiceAccount -Identity NazwaKonta
```

Przykład:

```
True
```

`True` potwierdza, że serwer, na którym wykonujemy test, potrafi pobrać hasło konta z AD — jeśli usługa mimo to nie startuje, przyczyna leży już nie w samym gMSA, tylko w konfiguracji usługi albo w uprawnieniu „Log on as a service".

Dodatkowo sprawdzamy, czy usługa faktycznie działa na tym koncie:

```
Get-Service NazwaUslugi | Select-Object Name, Status, StartType
Get-CimInstance Win32_Service -Filter "Name='NazwaUslugi'" | Select-Object StartName
```

Wycofanie zmian

Odinstalowanie konta z serwera, jeśli usługę przenosimy na inną maszynę albo wracamy do zwykłego konta domenowego:

```
Uninstall-ADServiceAccount -Identity NazwaKonta
```

Usunięcie samego konta z AD, jeśli nie jest już używane przez żaden serwer:

```
Remove-ADServiceAccount -Identity NazwaKonta
```

Gdy coś nie zadziała

Objaw	Przyczyna	Naprawa
`Test-ADServiceAccount` zwraca `False`	serwer nie jest członkiem grupy z `PrincipalsAllowedToRetrieveManagedPassword`, albo członkostwo się jeszcze nie zreplikowało	sprawdzamy grupę, czekamy na replikację AD albo wymuszamy ją ręcznie, resetujemy bilet Kerberos (`klist purge`)
`Add-KdsRootKey` / `Install-ADServiceAccount` działa dopiero po kilku godzinach	domyślny czas propagacji klucza KDS w lesie	używamy `-EffectiveTime (Get-Date).AddHours(-10)` przy tworzeniu klucza, jak w kroku 1
Usługa nie startuje mimo poprawnego `Test-ADServiceAccount`	brak uprawnienia „Log on as a service" dla konta	dodajemy konto w `secpol.msc` albo w odpowiednim GPO
`Install-ADServiceAccount` zwraca błąd dostępu	brak zainstalowanego modułu `RSAT-AD-PowerShell`, albo konto wykonujące instalację nie ma uprawnień do odczytu obiektu w AD	instalujemy moduł, sprawdzamy uprawnienia delegacji w OU, gdzie leży konto gMSA

Powiązane: RemoteApps w Remote Desktop Services

Źródła: notatki własne, na podstawie „Secure group managed service accounts" — Microsoft Learn

---

Revision #39
Created 2025-03-13 11:09:55 UTC by Tomasz Konieczny
Updated 2026-10-18 23:01:43 UTC by Tomasz Konieczny