



Active Directory Enumeration with AD Module without RSAT or Admin Privileges

This lab shows how it is possible to use Powershell to enumerate Active Directory with Powershell's `Active Directory` module on a domain joined machine that does not have Remote Server Administration Toolkit (RSAT) installed on it. Installing RSAT requires admin privileges and is actually what makes the AD Powershell module available and this lab shows how to bypass this obstacle.

Execution

The secret to being able to run AD enumeration commands from the AD Powershell module on a system without RSAT installed, is the DLL located in


`C:\Windows\Microsoft.NET\assembly\GAC_64\Microsoft.ActiveDirectory.Management` on a system that **has the RSAT** installed:

v4.0_6.3.0.0_31bf3856ad364e35

Share

View

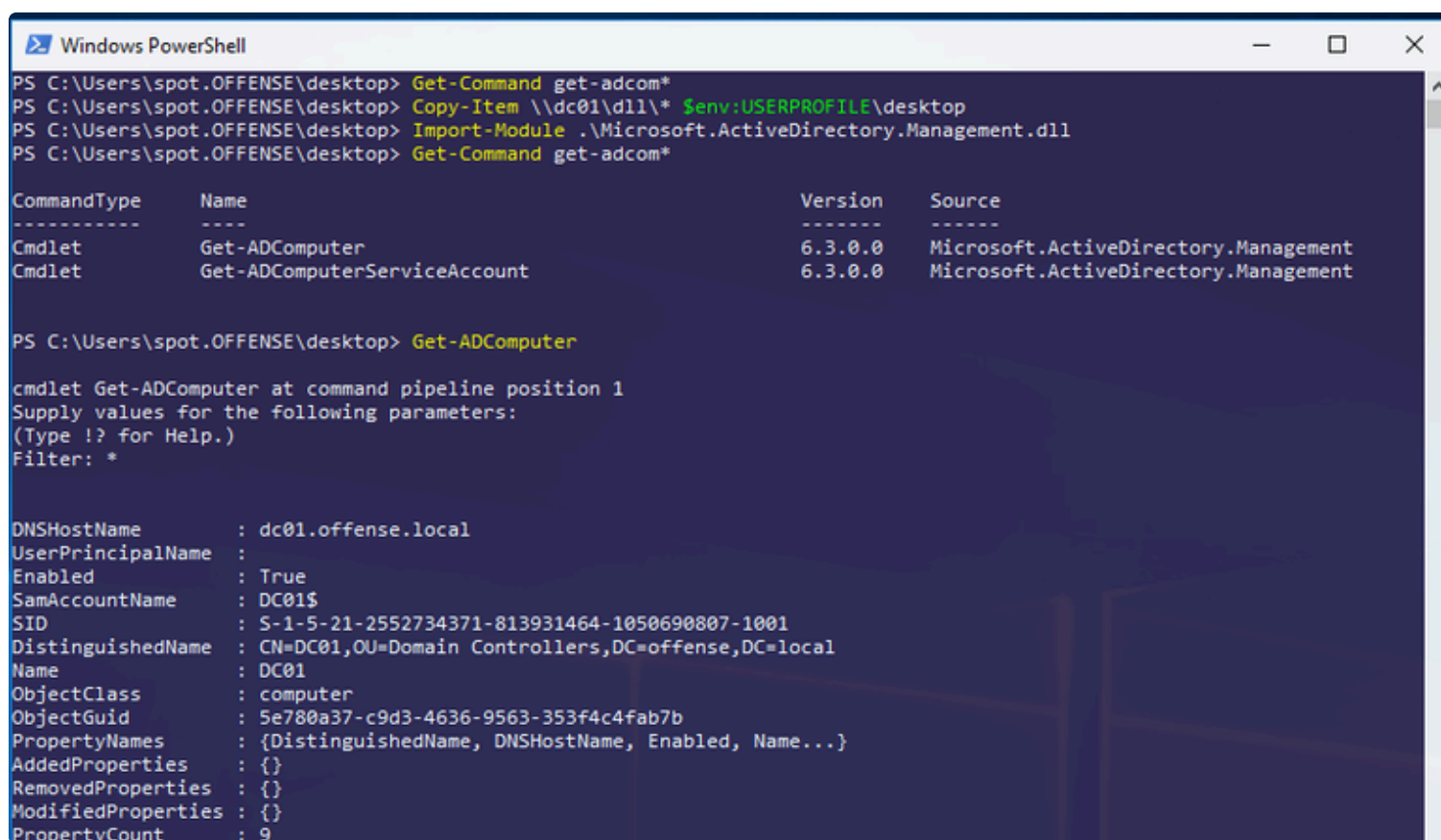
C:\Windows\Microsoft.NET\assembly\GAC_64\Microsoft.ActiveDirectory.Management\v4.0_6.3.0.0_31bf3856ad364e35

Name	Date modified	Type	Size
 Microsoft.ActiveDirectory.Management.dll	10/27/2018 6:43 PM	Application extens...	1,164 KB

This means that we can just grab the DLL from the system with RSAT and drop it on the system we want to enumerate from (that does not have RSAT installed) and simply import that DLL as a module:

```
Import-Module .\Microsoft.ActiveDirectory.Management.dll
```

Note how before we import the module, `Get-Command get-adcom*` returns nothing, but that changes once we import the module:



```
Windows PowerShell
PS C:\Users\spot.OFFENSE\desktop> Get-Command get-adcom*
PS C:\Users\spot.OFFENSE\desktop> Copy-Item \\dc01\dll\* $env:USERPROFILE\desktop
PS C:\Users\spot.OFFENSE\desktop> Import-Module .\Microsoft.ActiveDirectory.Management.dll
PS C:\Users\spot.OFFENSE\desktop> Get-Command get-adcom*

CommandType      Name                                     Version      Source
-----
Cmdlet            Get-ADComputer                         6.3.0.0      Microsoft.ActiveDirectory.Management
Cmdlet            Get-ADComputerServiceAccount           6.3.0.0      Microsoft.ActiveDirectory.Management

PS C:\Users\spot.OFFENSE\desktop> Get-ADComputer

cmdlet Get-ADComputer at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
Filter: *

DNSHostName       : dc01.offense.local
UserPrincipalName :
Enabled           : True
SamAccountName    : DC01$
SID               : S-1-5-21-2552734371-813931464-1050690807-1001
DistinguishedName : CN=DC01,OU=Domain Controllers,DC=offense,DC=local
Name              : DC01
ObjectClass       : computer
ObjectGuid        : 5e780a37-c9d3-4636-9563-353f4c4fab7b
PropertyNames     : {DistinguishedName, DNSHostName, Enabled, Name...}
AddedProperties    : {}
RemovedProperties  : {}
ModifiedProperties : {}
PropertyCount     : 9
```

As mentioned earlier, this does not require the user have admin privileges:

```
Windows PowerShell
PS C:\Users\spot.OFFENSE\desktop> net user spot /domain
The request will be processed at a domain controller for domain offense.local.

User name                spot
Full Name                spot
Comment
User's comment
Country/region code      000 (System Default)
Account active           Yes
Account expires          Never

Password last set        03/01/2019 20:10:52
Password expires         Never
Password changeable      03/01/2019 20:10:52
Password required        Yes
User may change password Yes

Workstations allowed     All
Logon script
User profile
Home directory
Last logon               03/02/2019 14:17:16

Logon hours allowed      All

Local Group Memberships
Global Group memberships *Domain Users
The command completed successfully.

PS C:\Users\spot.OFFENSE\desktop>
```

Download Management.DLL



1MB


Microsoft.ActiveDirectory.Management.dll

Microsoft.ActiveDirectory.Management.dll

Reference

<https://scriptdotsh.com/index.php/2019/01/01/active-directory-penetration-dojo-ad-environment-enumeration-1/>
scriptdotsh.com






Previous

Backdooring AdminSDHolder
for Persistence

Next

Enumerating AD Object Permissions
with dscls



Last updated 5 years ago