

Uruchamianie poleceń zdalnych info

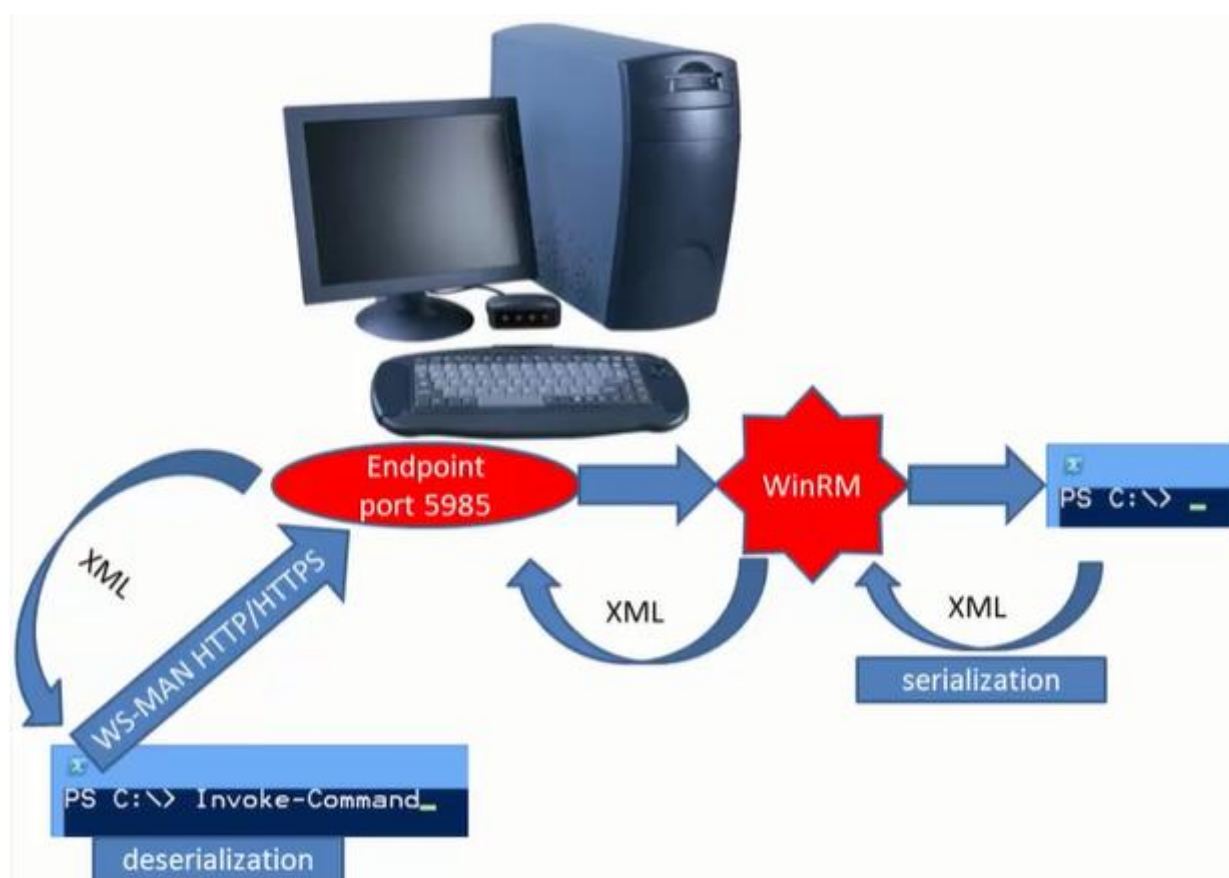
Wprowadzenie do remotigu, czyli trochę teorii

Uruchamianie komend zdalnych

Sesje interaktywne - polecenie Enter-PSSession

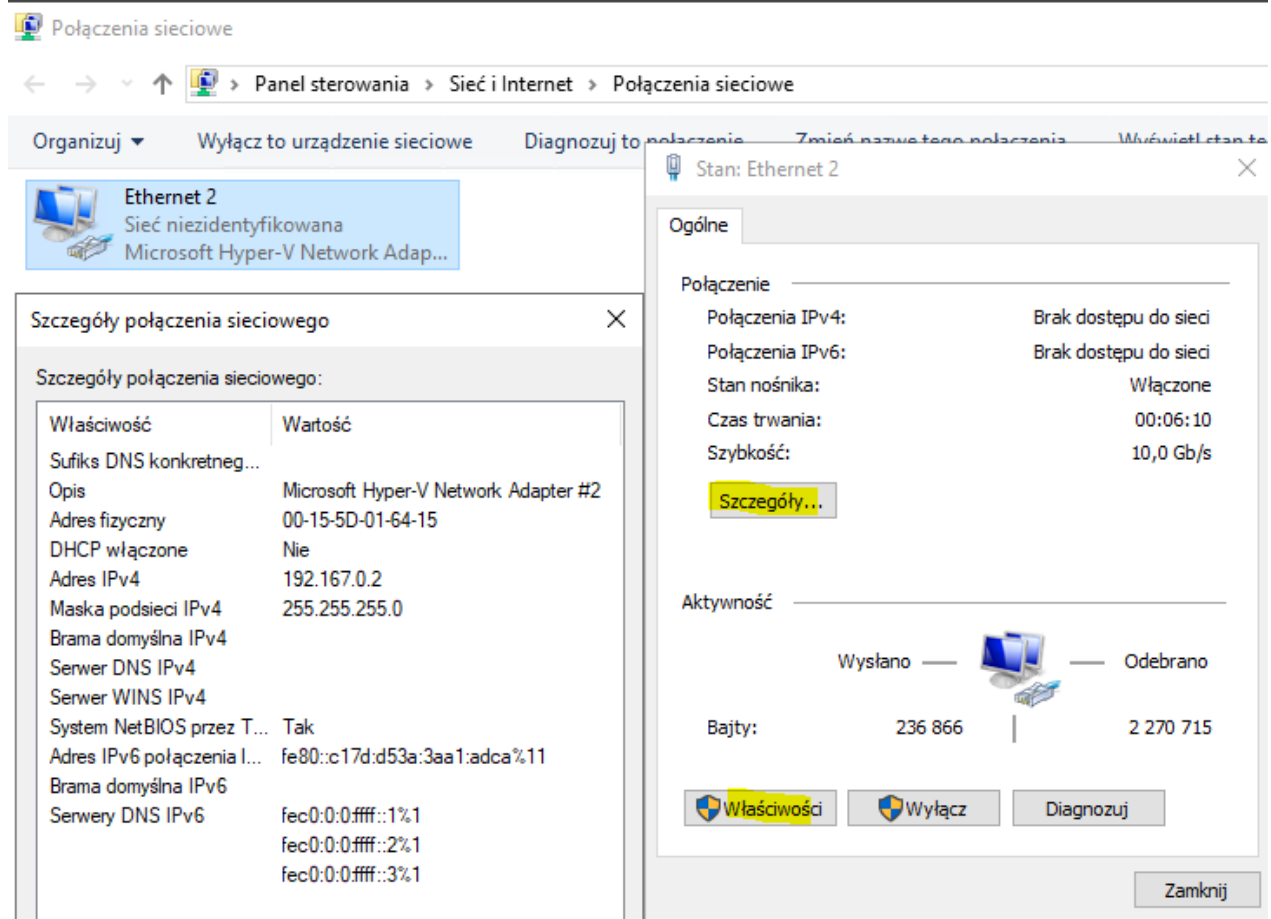
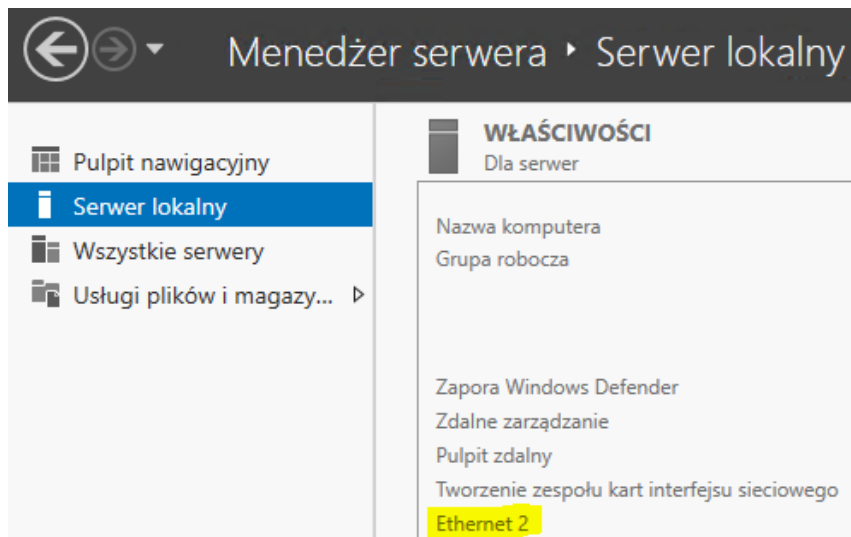
Polecenie wykonania zdalnego komendy - Invoke-Command

Co to jest serializacja i deserializacja

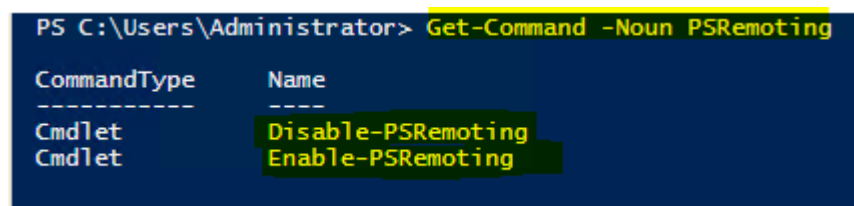


Obiekt, który został odtworzony po stronie klienta nie jest dokładnie takim samym klientem, który powstał na systemie zdalnym, większość właściwości przenosi się dokładnie tak samo zobaczycie numer procesu, zdalny numer procesu, ilość pamięci zajmowanej na zdalnym komputerze, ale obiekt straci swoje metody, ale można uruchomić te metody zdalnie.

Ustawiam konfigurację na komputerze zdalnym Windows 2019

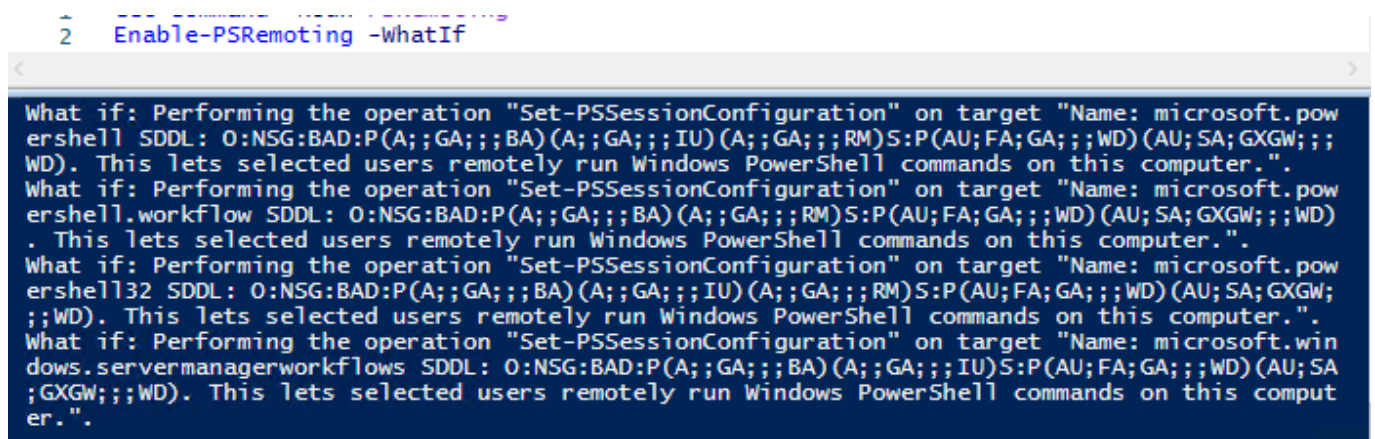


Remotig polega na przesłaniu do komputera zdalnego komend PowerShell które mają zostać tam uruchomione.



Sprawdzam co to polecenie zrobi

```
2 Enable-PSRemoting -WhatIf
```



Włączę PSRemoting

-SkipNetworkProfileCheck – wykona nawet gdy interfejs pracuje na profilu publicznym konfiguracji sieci

Wykonam Enable-PSRemoting -Force

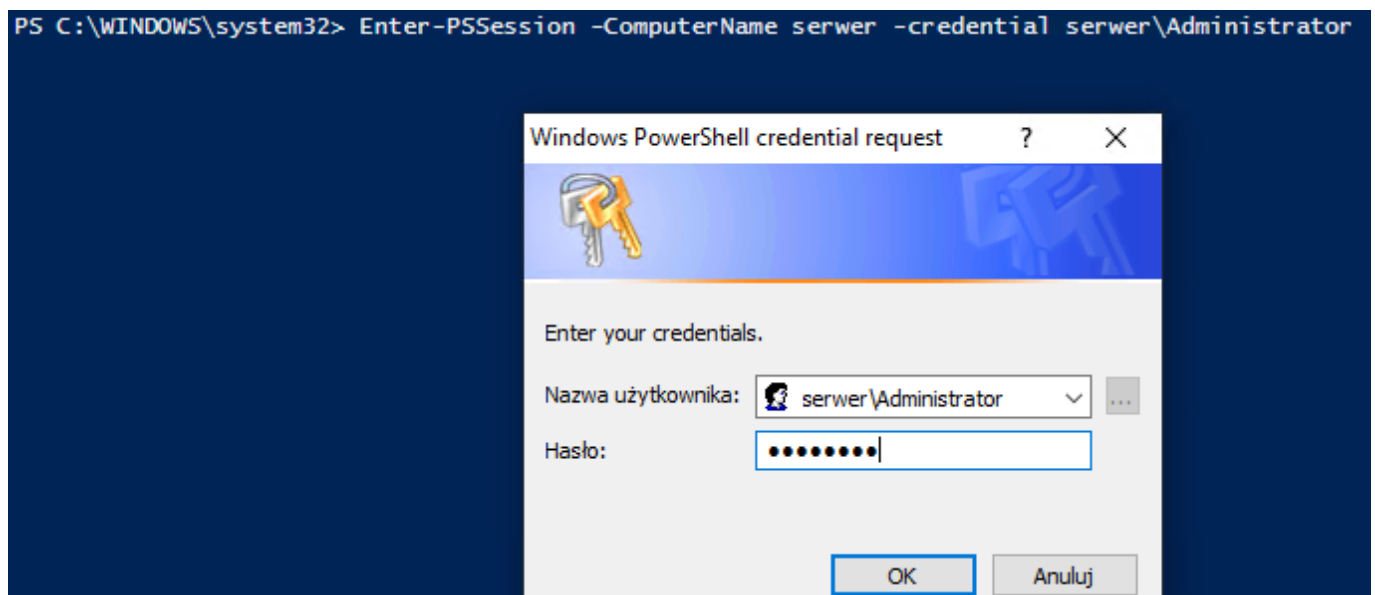
```
PS C:\Users\Administrator> Enable-PSRemoting -Force
PS C:\Users\Administrator>
```

DHCP włączone	Nie
Adres IPv4	192.167.0.21
Maska podsieci IPv4	255.255.255.0
Brama domyślna IPv4	192.167.0.2

Set-Item WSMan:\localhost\Client\TrustedHosts -Value "*" -Force

```
PS C:\WINDOWS\system32> Set-Item WSMan:\localhost\Client\TrustedHosts -Value "*" -Force
```

Enter-PSSession -ComputerName server -credential server\Administrator



```
PS C:\WINDOWS\system32> Enter-PSSession -ComputerName server -credential server\Administrator
[server]: PS C:\Users\Administrator\Documents>
```

Wszystkie polecenia działają na systemie zdalnym

```
4 Get-WmiObject -Class Win32_operatingSystem | FL *
```

```
PS C:\WINDOWS\system32> Enter-PSSession -ComputerName server -credential server\Administrator
[server]: PS C:\Users\Administrator\Documents> Get-WmiObject -Class Win32_operatingSystem | FL *
```

```
PSComputerName      : SERVER
Status              : OK
Name                : Microsoft Windows Server 2019 Standard|C:\Windows|\Device\Harddisk0\Parti
                    : tion1
FreePhysicalMemory   : 594548
FreeSpaceInPagingFiles : 1292320
FreeVirtualMemory    : 1985332
__GENUS             : 2
__CLASS              : Win32_OperatingSystem
__SUPERCLASS         : CIM_OperatingSystem
__DYNASTY             : CIM_ManagedSystemElement
__RELPATH             : Win32_OperatingSystem=@
__PROPERTY_COUNT      : 64
__DERIVATION          : {CIM_OperatingSystem, CIM_LogicalElement, CIM_ManagedSystemElement}
__SERVER              : SERVER
__NAMESPACE          : root\cimv2
__PATH               : \\SERVER\root\cimv2:Win32_OperatingSystem=@
BootDevice           : \Device\HarddiskVolume1
BuildNumber          : 17763
BuildType            : Multiprocessor Free
Caption              : Microsoft Windows Server 2019 Standard
CodeSet              : 1250
CountryCode          : 48
CreationClassName     : Win32_OperatingSystem
CSCreationClassName  : Win32_ComputerSystem
CSDVersion            :
CSName               : SERVER
```

Do zakończenia sesji służy

```
[server]: PS C:\Users\Administrator\Documents> Exit-PSSession
PS C:\WINDOWS\system32>
```

Drugi sposób pracy to wysyłanie z komputera lokalnego do zdalnego pojedynczych komend.

`Invoke-Command -ComputerName server -credential server\Administrator -ScriptBlock { Get-WmiObject -Class Win32_operatingSystem }`

```
PS C:\WINDOWS\system32> Invoke-Command -ComputerName server -credential server\Administrator -ScriptBlock { Get-WmiObject -Class Win32_operatingSystem }
```

```
SystemDirectory : C:\Windows\system32
Organization     :
BuildNumber      : 17763
RegisteredUser   : Użytkownik systemu Windows
SerialNumber     : 00429-00104-70082-AA484
Version          : 10.0.17763
PSComputerName   : server
```

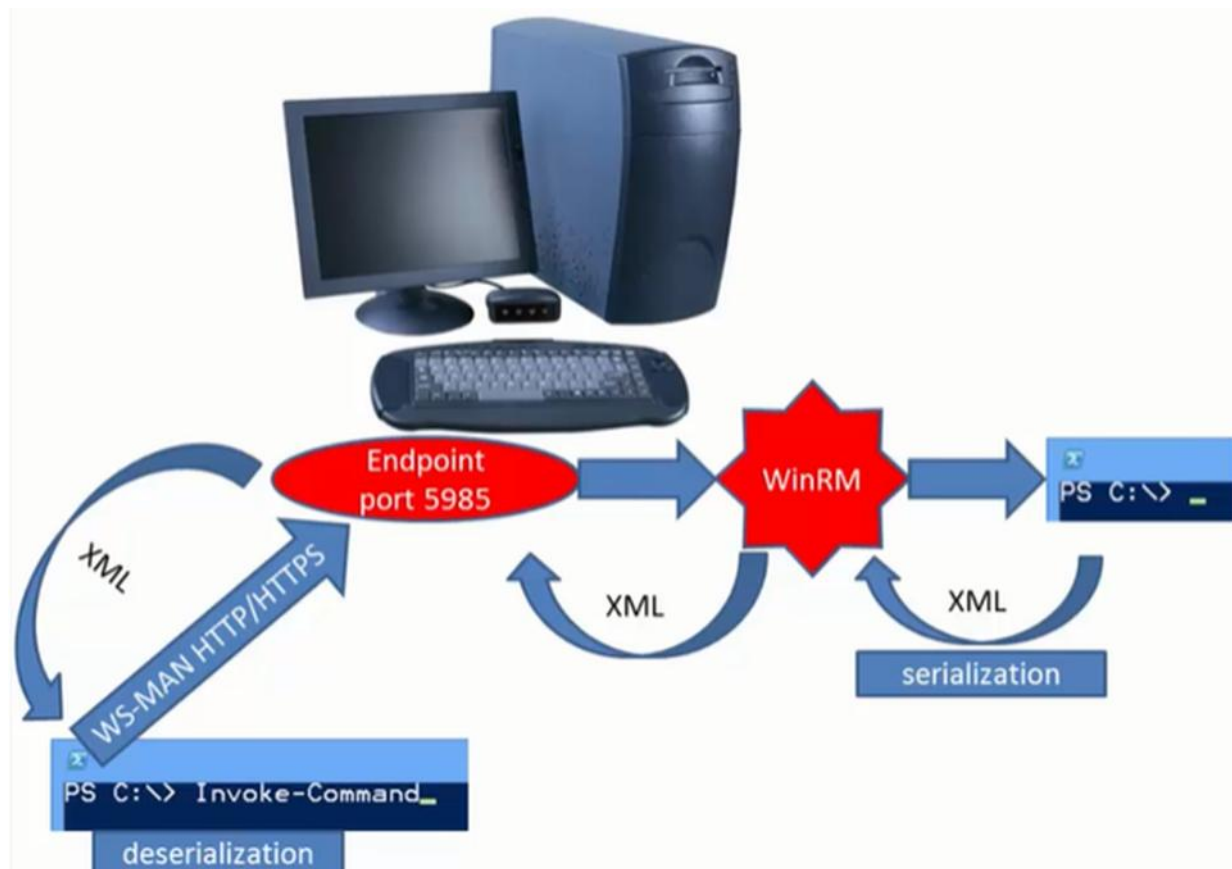
```
SystemDirectory : C:\Windows\system32
Organization     :
BuildNumber      : 17763
RegisteredUser   : Użytkownik systemu Windows
SerialNumber     : 00429-00104-70082-AA484
Version          : 10.0.17763
PSComputerName   : server
```

Może być wykorzystywany w skryptach, bo nie jest interaktywna sesją z komputerem zdalnym

Jakiego typu obiekt jest zwracany, gdy polecenie jest wykonywane na komputerze zdalnym

```
Invoke-Command -ComputerName serwer -credential serwer\Administrator -ScriptBlock { Get-WmiObject -Class Win32_operatingSystem } | Get-Member
```

```
TypeName: Deserialized.System.Management.ManagementObject#root\cimv2\Win32_OperatingSystem
```



Jakiego typu obiekt jest zwracany, gdy polecenie jest wykonywane na komputerze lokalnym

```
PS C:\WINDOWS\system32> Get-WmiObject -Class Win32_operatingSystem | Get-Member
```

```
TypeName: System.Management.ManagementObject#root\cimv2\Win32_OperatingSystem
```

Poznałeś trochę teorii na temat remotingu

Zobaczyłeś, jak uruchamiać komendy zdalne

Wiesz, jak pracować z sesją interaktywną inicjowaną Enter-PSSession

Wiesz jak wysyłać polecenia zdalne korzystając z Invoke-Command

Wiesz co to jest serializacja i deserializacja