

# The CIM Cmdlets

---



**Jeff Hicks**

AUTHOR ~ TEACHER ~ SENSEI

@jeffhicks <https://jdhitsolutions.com/blog>



# CIM?

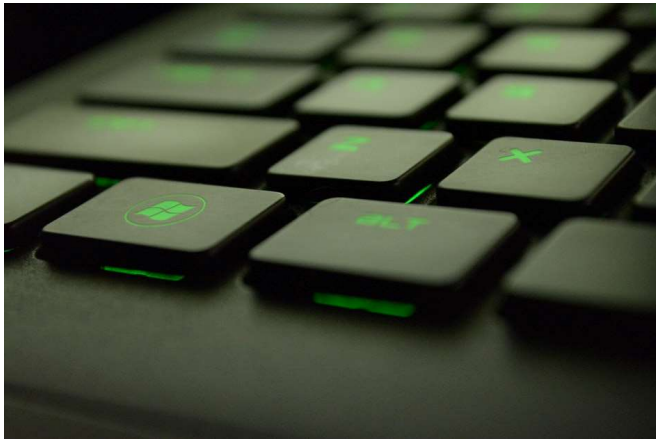
**Common Information Model**

**Based on industry develop  
standards and protocols**

**WMI is Microsoft's  
implementation**

**Was a great idea 20 years  
ago...**



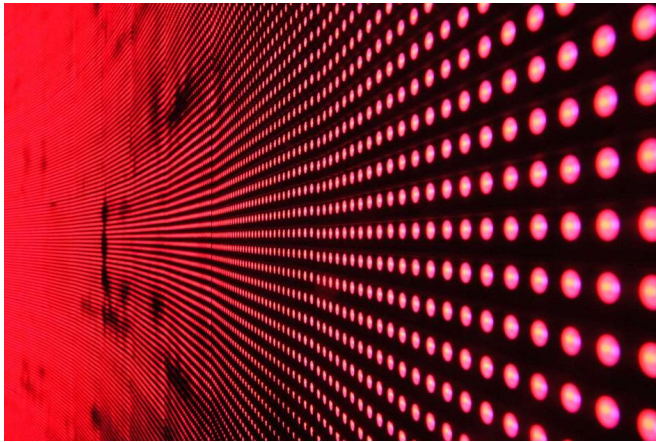


WMI cmdlets rely on legacy protocols

Not firewall friendly

Rely on proprietary Microsoft code





**Microsoft now embracing open standards**

**Use of WSMAN for remoting**

- Full SSH support on the horizon

**Implementing CIM based standards**

- Support for non-Microsoft platforms



# CIM Cmdlets

**Query the same  
WMI information**

**Uses PowerShell  
remoting  
(WSMan)**

**Firewall friendly**



# WMI vs CIM Commands

## WMI

```
Get-Wmiobject -Class  
win32_service -Filter  
"name='winrm'" -ComputerName srv1
```

```
Get-WmiObject -list win32_bios
```

```
Get-WmiObject win32_process  
-filter "name='notepad.exe'" |  
Invoke-WmiMethod -name Terminate
```

## CIM

```
Get-CimInstance -ClassName  
win32_service -Filter  
"name='winrm'" -ComputerName srv1
```

```
Get-CimClass win32_bios
```

```
Get-CimInstance win32_process  
-filter "name='notepad.exe'" |  
Invoke-CimMethod -Name Terminate
```





# Limitations

## **Restricted by WSMAN limitations**

- CIM Session may provide solution

## **CIM cmdlets lack direct support for alternate credentials**

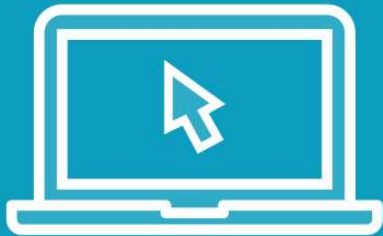
- Requires a CIM Session

## **Requires PowerShell 3.0**

- CIM cmdlets can be configured to support legacy (i.e. DCOM) systems



# Demo



## Remote systems management with CIM and PowerShell





# Summary



**CIM Cmdlets are the future for remote management**

- Firewall friendly
- Cross-platform support
- No proprietary Microsoft “stuff”

**Replace WMI commands with the CIM cmdlets**

**Create your own tools based on CIM cmdlets and WMI classes**

