

PowerShell and the Web



Jeff Hicks

AUTHOR ~ TEACHER ~ SENSEI

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





PowerShell is designed with the cloud and Internet in mind

- This could be an internal server or platform

PowerShell commands can abstract the process

May require a bit of scripting



Invoke-WebRequest <http://www.icanhazip.com> | Select Content

Invoke-WebRequest

Send http and other requests to a web site

Response is a structured object



```
Invoke-WebRequest http://www.icanhazip.com | Select Content  
Invoke-WebRequest -Uri $uri -OutFile $out -DisableKeepAlive  
-UseBasicParsing
```

Invoke-WebRequest

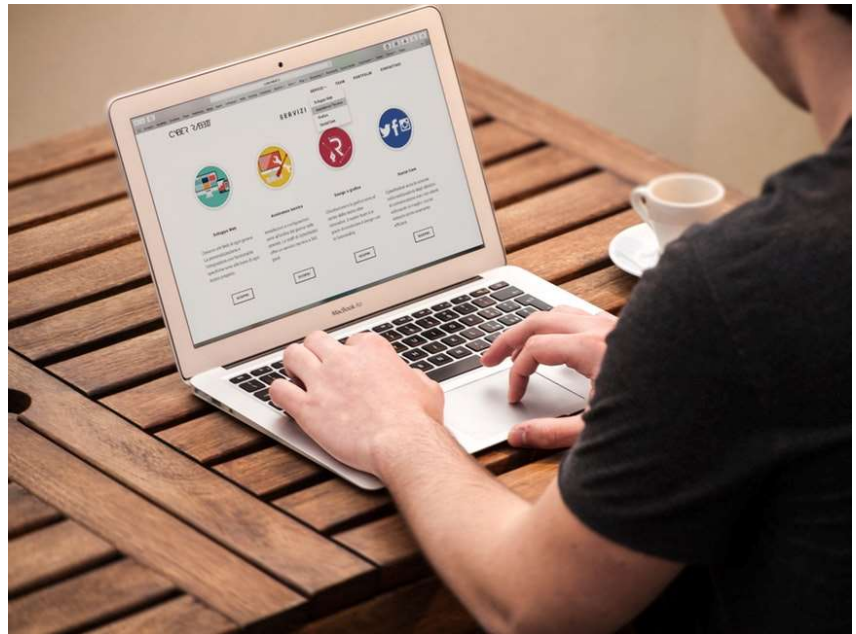
Send http and other requests to a web site

Response is a structured object

Handy for “scraping” a site or downloading



This is not intended to
automate your
browser



```
$uri = 'https://api.github.com/users/jdhitsolutions/repos'  
invoke-restmethod "$uri`?per_page=50" -Method Get
```

Invoke-RestMethod

Send requests to REST services

(Usually) automatically converts JSON to objects

Also handy for structured data like RSS XML



```
PS C:\>  
PS C:\> get-rssfeed http://jdhitsolutions.com/blog/feed
```

```
Title      : PowerShell Pop Quiz  
Author     : Jeffery Hicks  
Description: I'm always looking for ways to help you learn PowerShell. I want to have PowerShell  
            teach you itself? I have created a PowerShell blog on cmdlets and  
            functions installed on your computer. In short, I want to help you learn PowerShell. I  
            presents a [...]  
Published  : 9/13/2017 1:02:29 PM  
Category   : {PowerShell, Training, Scripting}  
Link       : http://jdhitsolutions.com/blog/powershell/5641/powershell-pop-quiz/#utm_source=feed&utm_medium=feed&utm_campaign=feed
```

```
Title      : Get Git with PowerShell  
Author     : Jeffery Hicks  
Description: If you are creating PowerShell scripts, tools or modules today, you are most likely using Git. What?  
            You're not? Is it because you haven't gotten around to installing it? I have some "quick and dirty"  
            PowerShell hacks to help you out on Windows systems. Linux boys and girls already know what to do.  
            Naturally you [...]  
Published  : 8/25/2017 12:25:24 PM  
Category   : {Git, PowerShell, Scripting}  
Link       : http://jdhitsolutions.com/blog/powershell/5633/get-git-with-powershell/#utm_source=feed&utm_medium=feed&utm_campaign=feed
```

```
Title      : Enhancing PSVersionTable  
Author     : Jeffery Hicks  
Description: Not too long ago I posted a PowerShell function that could provide detail about the PowerShell engine  
            driving your current PowerShell session. I like having a function that writes an object to the pipeline,  
            can take parameters and offer help documentation. But there's an alternative approach you could also  
            take. Most of the information is [...]  
Published  : 8/24/2017 12:34:04 PM  
Category   : {PowerShell, PSLinux, psversiontable}  
Link       : http://jdhitsolutions.com/blog/powershell/5616/enhancing-psversiontable/#utm_source=feed&utm_medium=feed&utm_campaign=feed
```

This is my function
wrapped around Invoke-
RestMethod

```
$px = new-webserviceproxy -uri  
http://192.168.3.51/mywebservices/firstservice.asmx
```

New-WebServiceProxy

Web services are commands called by other web applications

But you can create a proxy object




```
$px = new-webserviceproxy -uri  
http://192.168.3.51/mywebservices/firstservice.asmx  
$px.Add(123,321)
```

New-WebServiceProxy

Web services are commands called by other web applications

You can create a proxy object

Invoked methods are passed on to the web service



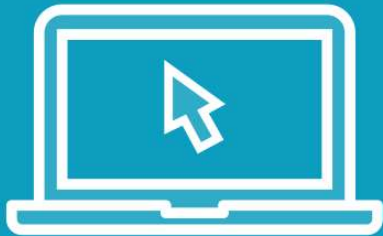
Web Services not as
widely used

RESTful APIs and
Odata seem to be
taking over

But you might have an
older internal system
offering web services



Demo



PowerShell and the Web



Advanced Applications



oData and PowerShell

Open Data Protocol

Open standard for consuming RESTful APIs

<http://odata.org>



Export-oDataEndPointProxy

```
$name = "odatademo"  
$base = "C:\Program Files\WindowsPowerShell\Modules\  
$outpath = Join-path -Path $base -ChildPath $Name  
mkdir $outpath
```

```
Export-ODataEndpointProxy -Uri  
http://services.odata.org/OData/OData.svc/  
-AllowUnsecureConnection -OutputModule $outpath -force
```



```
Administrator: Windows PowerShell 5.1.16296

PS C:\> get-module odatademo -ListAvailable

Directory: C:\Program Files\WindowsPowerShell\Modules

ModuleType Version      Name                                ExportedCommands
-----
Manifest    1.0             odatademo                        {Get-Product, New-Product, Set-Product, Remove-Product...}

PS C:\> get-command -Module odatademo | group noun

Count Name                                     Group
-----
4 Advertisement {Get-Advertisement, New-Advertisement, Remove-Advertisement, Set-Advertisement}
4 Category     {Get-Category, New-Category, Remove-Category, Set-Category}
4 Person       {Get-Person, New-Person, Remove-Person, Set-Person}
4 PersonDetail {Get-PersonDetail, New-PersonDetail, Remove-PersonDetail, Set-PersonDetail}
4 Product      {Get-Product, New-Product, Remove-Product, Set-Product}
4 ProductDetail {Get-ProductDetail, New-ProductDetail, Remove-ProductDetail, Set-ProductDetail}
4 Supplier     {Get-Supplier, New-Supplier, Remove-Supplier, Set-Supplier}

PS C:\> get-person -AllowUnsecureConnection -OrderBy Name

ID Name
--
2 Art Braunschweiger
6 Fran Wilson
5 Jaime Yorres
1 Jose Pavarotti
4 Liu Wong
3 Liz Nixon
0 Paula Wilson
```

Advanced Applications



PSSwagger

Auto generate cmdlets from RESTful services using Swagger and Open API specs

Heavy developer focus

<https://github.com/PowerShell/PSSwagger>



Summary



More information is being made available via the Internet and related services

Using these cmdlets for products with no PowerShell support

You will want to build tooling around these cmdlets

