



# PowerShell Survival Guide

André Melancia  
Ricardo Cabral

# Datacenter Airlift



28 Novembro \ Microsoft Portugal

# > Get-About "Ricardo Cabral"

## ⇒ Contacts:

- ⇒ rramoscabral@gmail.com
- ⇒ <https://LinkedIn.COM/in/rrcabral>
- ⇒ <https://Facebook.COM/ricardo.cabral.pt>
- ⇒ <https://Twitter.COM/rrramoscabral>

## ⇒ IT Manager at Globaltemp

## ⇒ Degree in Computer Engineering

## ⇒ Actively participating, volunteering and/or speaking

- ⇒ Portuguese community meetings (PowerShell Portugal, IT Pro Portugal, Arduino/Genuino Day, AzurePT, Netponto, O365PT, PTJUG, PTXUG, SQLPort, SQLSaturday Lisboa & Porto, Tuga IT, etc.)

## ⇒ Passionate and self-taught in information technology with over 13 years' of experience in IT management, development and projects.

## ⇒ Loves sharing, living and learning.



**Microsoft**

Specialist

Developing Microsoft  
Azure Solutions



# > Get-NSASpyInfo "André Melancia"

- ➔ Developer / DBA
- ➔ Microsoft Certified Trainer (MCT)
- ➔ Food devouring expert
- ➔ <http://Andy.PT>
  - ➔ [Andy@Andy.COM.PT](mailto:Andy@Andy.COM.PT)
  - ➔ <https://LinkedIn.COM/in/AndreMelancia>
  - ➔ <https://Facebook.COM/Andy.COM.PT>
  - ➔ <https://Twitter.COM/AndyPT>



## ➔ Volunteer and/or Speaker at:

PowerShell Portugal, IT Pro Portugal, SQL Saturdays, SQLBits, SQLRelay, Tuga IT, 24HoP, PASS Global Portuguese VC, IoT Portugal, SQLPort, Global Azure Bootcamp Lisbon, Arduino Day Lisbon, NetPonto, PTXug, PHPLx, EuroDIG, IPv6 Portugal, DNSSec Portugal, etc.

# > Get-CommunityContacts

## ➔ PowerShell Portugal

- ➔ FB Página: <http://Facebook.COM/PowerShellPortugal>
- ➔ FB Grupo: <http://Facebook.COM/groups/PowerShellPortugal>
- ➔ Twitter: <http://Twitter.COM/PowerShellPT>

## ➔ IT Pro Portugal

- ➔ FB Página: <http://Facebook.COM/ITProPortugal>
- ➔ FB Grupo: <http://Facebook.COM/groups/ITProPortugal>
- ➔ Twitter: <http://Twitter.COM/ITProPortugal>

## ➔ Lista de eventos de comunidades

- ➔ <http://Facebook.COM/groups/WindowsDevPT> (lista pinned on top)



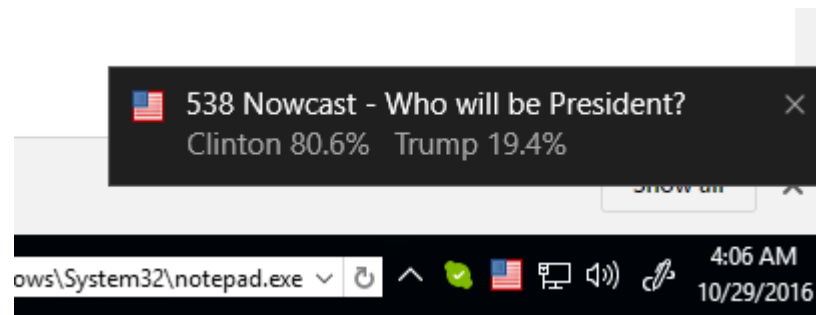
# PowerShell?



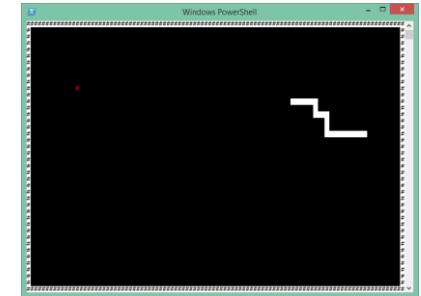
Why the hell?

# > Get-PSExamples

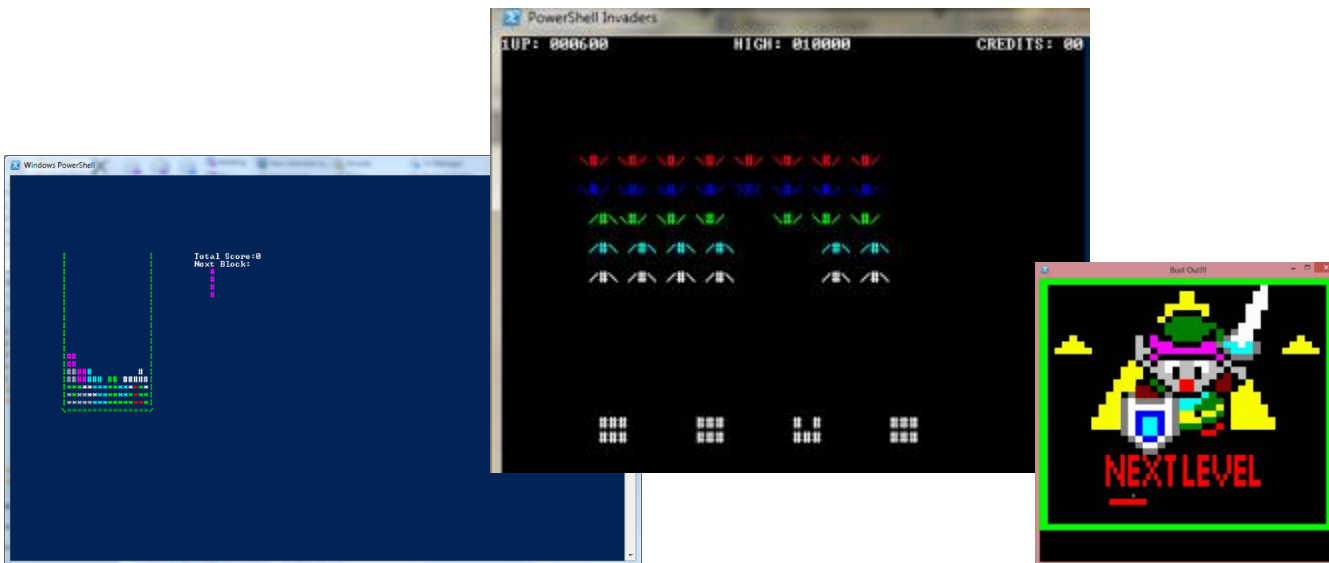
→ <https://github.com/ctrlbold/nowcastnotify>



→ <http://www.twentysidedblog.com/a-powershell-snake-game/>

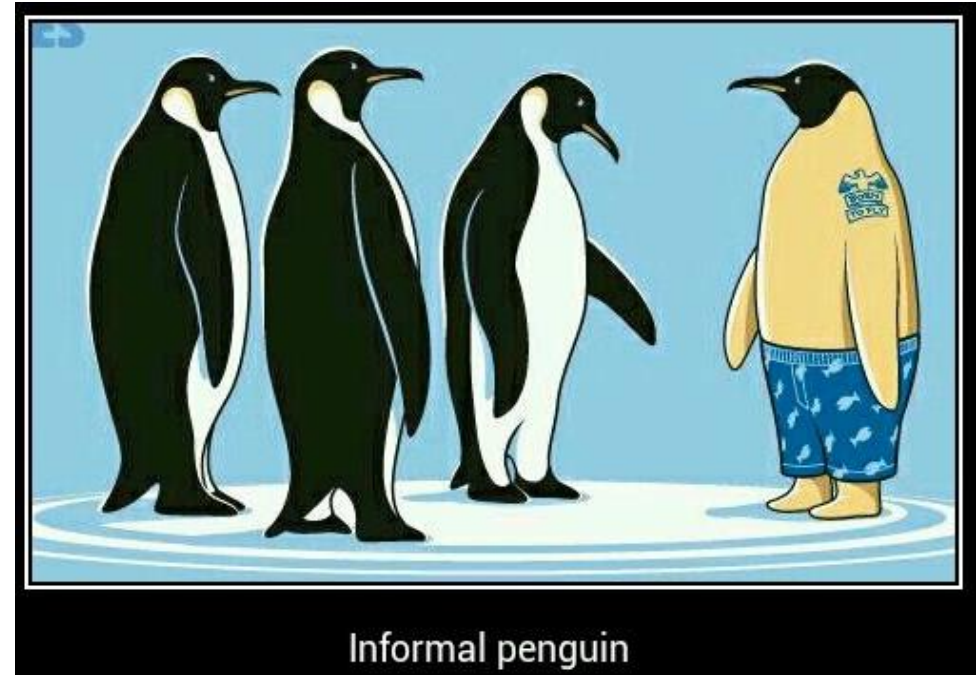


→ <https://blogs.technet.microsoft.com/mspfe/2012/09/10/how-to-create-video-games-using-powershell>



# > Open Source

- Accepts bug corrections and contributions from the community
- .Net Framework open source!
- Bash on Windows
- PowerShell open source!
  - <https://github.com/PowerShell>
- PowerShell on Linux and MacOS
  - <https://github.com/PowerShell/PowerShell/releases>



# > Install PowerShell on Linux

## ➔ Get the files!

➔ <https://github.com/PowerShell/PowerShell/releases>

## ➔ RedHat family (RHEL, CentOS, etc.)

➔ Download "`*centos.x86_64.rpm`"

➔ "`sudo yum install /path/to/*.rpm`"

➔ Pre-requisites installed automatically (7.2 anyway)

## ➔ Debian family (Debian, Ubuntu, etc.)

➔ Version dependent pre-requisites!

➔ Ubuntu 16.\* [or 14.\*]:

➔ "`sudo apt-get install libunwind8 libicu55`" [or "`libicu52`"]

➔ "`sudo dpkg -i /path/to/*.deb`"

**Global warming,  
my ass.  
I'm freezin'!**





```
[Andy@AndyDINO POWERSHELL]$ sudo -s
[root@AndyDINO POWERSHELL]# dir
powershell-6.0.0_alpha.9-1.e17.centos.x86_64.rpm
[root@AndyDINO POWERSHELL]# yum install powershell-6.0.0_alpha.9-1.e17.centos.x86_64.rpm
Loaded plugins: fastestmirror, langpacks
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
Examining powershell-6.0.0_alpha.9-1.e17.centos.x86_64.rpm: powershell-6.0.0_alpha.9-1.x86_64
Marking powershell-6.0.0_alpha.9-1.e17.centos.x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package powershell.x86_64 0:6.0.0_alpha.9-1 will be installed
--> Processing Dependency: libunwind for package: powershell-6.0.0_alpha.9-1.x86_64
base | 3.6 kB | 00:00:00
extras | 3.4 kB | 00:00:00
openlogics | 1.3 kB | 00:00:00
updates | 3.4 kB | 00:00:00
(1/3): openlogics/7/x86_64/primary | 9.6 kB | 00:00:00
(2/3): extras/7/x86_64/primary_db | 160 kB | 00:00:00
(3/3): updates/7/x86_64/primary_db | 6.4 MB | 00:00:00
Determining fastest mirrors
openlogics 48/48
--> Processing Dependency: libicu for package: powershell-6.0.0_alpha.9-1.x86_64
--> Running transaction check
---> Package libicu.x86_64 0:50.1.2-15.e17 will be installed
---> Package libunwind.x86_64 2:1.1-5.e17_2.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
powershell x86_64 6.0.0_alpha.9-1 /powershell-6.0.0_alpha.9-1.e17.centos.x86_64 121 M
Installing for dependencies:
libicu x86_64 50.1.2-15.e17 base 6.9 M
libunwind x86_64 2:1.1-5.e17_2.2 updates 56 k
=====

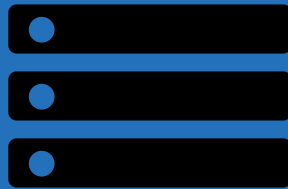
Transaction Summary
-----
Install 1 Package (+2 Dependent packages)

Total size: 128 M
Total download size: 6.9 M
Installed size: 145 M
Is this ok [y/d/N]:
```



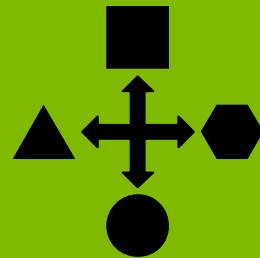
# Free Azure Trial

<http://aka.ms/tryazure>



## Try Windows Server 2016

<http://aka.ms/tryws2016>



## Try System Center 2016

<http://aka.ms/trysc2016>



## Try OMS

<http://aka.ms/tryoms>



> Get-PSDemos

~~ONLY TRY  
THIS AT  
HOME~~

**Try this  
anywhere!**



# > Receive-Questions –Input "Speech"



OBRIGADO!

THANK YOU!

Дуже дякую!

Go raibh maith agaibh!

Danke!

Diolch!

¡Gracias!

Arigato!

Merci!



André Melancia

<http://Andy.PT>



Ricardo Cabral

<https://blog.pt.rramoscabral.com>







```
1 <#
2 .NOTES
3 -----
4
5 Created on:      2015-09-21 09:05
6 Created by:      Arash Nabi
7 Email:           arash@nabi.nu
8
9 -----
10
11 .Synopsis
12 Send Daily Dilbert Cartoon from www.dilbert.com
13 .EXAMPLE
14 Today's Dilbert: Get-Dilbert
15 .EXAMPLE
16 Yesterday's Dilbert: Get-Dilbert -last -1
17 #>
18 function Get-Dilbert
19 {
20     [CmdletBinding()]
21     [OutputType([int])]
22     Param
23     (
24         [Parameter(ValueFromPipelineByPropertyName = $true,
25                     HelpMessage = "Choose last date",
26                     Position = 0)]
27         [ValidateSet ('-1',
28                     '-2',
29                     '-3',
30                     '-4')]
31         [int]$Last
32     )
33
34
35
36
37 Process
38 {
39     $VerbosePreference = "Continue"
40     if ($Last)
41     {
42         $lastDate = (get-date).AddDays($Last) | get-date -Format "yyyy-MM-dd"
43         Write-Verbose 'Invoking website'
44         $dil = Invoke-WebRequest -Uri "http://dilbert.com/strip/$LastDate" -UseBasicParsing
45         $bild = $dil.Images | Where-Object { $_.class -eq 'img-responsive img-comic' } | Select-Object src
46         Invoke-WebRequest $bild.src -OutFile "$PSScriptRoot\$LastDate.gif"
47         Write-Verbose 'Sending email'
48         Send-MailMessage -to Email.To@domain.com -From Email.From@domain.com -SmtpServer smtp.domain.com -Subject "Dilbert $LastDate" -BodyAsHtml "<br/
49     }
50     else
51     {
52         $today = (get-date -Format "yyyy-MM-dd")
53         Write-Verbose 'Invoking website'
54         $dil = Invoke-WebRequest -Uri "http://dilbert.com/strip/$today" -UseBasicParsing
55         $bild = $dil.Images | Where-Object { $_.class -eq 'img-responsive img-comic' } | Select-Object src
56         Invoke-WebRequest $bild.src -OutFile "$PSScriptRoot/$today.gif"
57         Write-Verbose 'Sending email'
58         Send-MailMessage -to Email.To@domain.com -From Email.From@domain.com -SmtpServer smtp.domain.com -Subject "Dilbert $today" -BodyAsHtml "<br/><i
59     }
60 }
61
62
63 }
64
65 }
```

