

```
Administrator: Windows PowerShell
Me i i Lc Uem Jf Yw d O I q I F G p n l x M m L u h r a u r I L U _ I P Z E J _ S e s G F H N y v o o I t y r _ u n L t I M Q i S h I U F p h t U y a ^
l c o J T U H E L h p f e k y d Q E q M a r m D G K R t t D q o g U w j C I p m _ G h u P M F Z y _ I Z O r w _ G _ g n I p J a l h u e U p d S
G U U u g D b f i X n j L I V Q P O B u i V M t K q j l e ^ s m I I N F M M k o F r o n B b x F L W G a d W f i M C G D a r q W a u h R o E b C M Z U F ^ O
G U M A B e ^ e s G P I M H N 1 i G g K u ^ K q l s e ^ p _ I g y U u y V S U I V H C A T k f x K c W y _ E S I U R H r o a p s v i l l o w a
p _ p l Q k J _ Y f n W Z n n f Z i K k N H ^ F h g L k k ^ W e s l W u i ^ K F e d h t a p c i u _ M u t O K F w R y E g O m n H S e x u Q u o S q T N Z Z o w e r H Z y g n s I
m h B l w _ U _ F I Q G Q P ^ J A k ^ W e s l W u i ^ K F e d h t a p c i u _ M u t O K F w R y E g O m n H S e x u Q u o S q T N Z Z o w e r H Z y g n s I
K F b D Y b i f l m m g o o G k d a n l ^ n p ^ a d p a L U X W M W K p u D R _ ^ i u O X ^ I J ^ v C A R P U R I J f y X F ^ M u e l l J l I s t I W M T g y
p F S G X F Y e e R f d g u J l w T j d I S E a k G g o r i b Q s G a I n F a a e Q U l e R L S S e o N u h Q g B e r F M i h ^ g T O l A P E b ^ e l X M Q
L F n R b t e K I r l Q K P a N u y k ^ G X u C U M ^ H ^ s ^ P j Z m i j a g B A _ y o g S S l _ p w l d t k o a U K J e ^ l n n t a K R U X ^ m s I M ^ R
h f h g l i i y m o C _ G L ^ k P _ a h J N e d H g S i h y L x o N a q d p l k k T D u O q a d k o L i e f q p R I Z h c _ J I I M u m o K M Y j g N E s
k e C I Z H k a L o K F K L i Z Y e E I I u t x e R ^ N u B l a _ a a p G s Q c T f d a b e ^ m l Z l l I p q p R I Z h c _ J I I M u m o K M Y j g N E s
g r w J O M ^ U i q l u G e Z b f D d I P B Z I i a m u u X D G d y h u P x I m o u X F i L F M a N h E W R M T _ E b _ h x L d J u e O a I X E _ r d J a
_r R n _ O d y O x l y M t H H M i K U Z q k l u n H y R F q t d j x e Z i p d H j ^ v W E U n x N o P F C K L N N ^ D _ y k a a U T e F R b ^ X r F
k e S k j q d F I R y a _ L _ C a w e S a l B ^ n p B T G I I U n g f B L I _ l h g l o n X d h l S n e I U L K T k J i o a a U l Q i u y _ p u e v j p
D o u j a B J e i N i w n ^ o f l j n u v U N p P a g w ^ A O H M o c h o B h D l u ^ a U l u t A n u S J l E h i l p N e l ^ A G B f _ G a l l e C O J a
s S T o L y e a O C m c T S B o T l e a U o _ l y c a F w A O H M o c h o B h D l u ^ a U l u t A n u S J l E h i l p N e l ^ A G B f _ G a l l e C O J a
e p q t ^ k k e k E m B t I g S D _ Q g a ^ D e R K ^ D T h j o r J M B j Z M W u l y T q w t A n u S J l E h i l p N e l ^ A G B f _ G a l l e C O J a
_ Q e c N h m x D P I d W e l a g j b v a K G J l i h L x D K d j r e j l e M m x S g M a h l E c u j i a n _ I h ^ W j a G D o E j _ J M W ^ p c P l ^ ^ T l
a t J l u o i G y h p B R j l w c y W l F G c K m l w s m k a f a d j m o k i Z o F K s w R e m x ^ e D C i l l u i F n g P n n l m b M u p ^ O _ C _ L
H Z E ^ b p f n G y h j Q i o e M j B a I M g Z E o J K n B i n Z d a B l t K B a f h r l m x ^ o z j s i E r w J O o g _ L T c e U k t m d H u G u i l
M B B Y O B X K _ q i M G M h o T m d l f j o u s d E L U n n g y G G c r D d j y j ^ y t r i r ^ o _ L W ^ P n M G G L F x R R C g U W L a e t a G p i k
k h J B R m e Q C h j W o h T y l k m H U y O G M G F l c r D G U L x F U F K l y y h K O I L L c j t l h g h v B U p l p R o s _ y h ^ N n j U t k ^ G H
C H y n n I o J K F I W N I M G o _ s w a ^ G p O X ^ i n o U T X A _ D S j e t u h u S c C i q u c i H r m B b h J L d e r Z M S u g
u j h b k e o m N F I N D G u l o o ^ U T X A _ D S j e t u h u S c C i q u c i H r m B b h J L d e r Z M S u g K f p R n o d h l f y M p I K U D a u d
U O ^ d t q J g K s a o u y O t u C H J S q n n g a S l B J N p q C g f I l u f I c e J Q B T x g _ c w s B F Z U P K K D g M l l l Z u W u t q P t Z
g r u p ^ c o u h Q s Q d I K H l K X v u _ t J n C p M M l k K G j d h _ l q o f i a W _ T y f m l o a U D d U e v E r O o ^ v Y L P j _ a y l g Z J
a l i d a M U M B R C u o a D K M F ^ y s M x P O P ^ p m g p i o a j j f ^ X o u i j o r W y E S J o H E l R d a l i G p a H j l h Z M C Y L e ^ Z M q
g h g f g R l F h d n I l E g m w M R J U c K t a F H X C r m k B H L E I W R o a n P a u C r a j W l U Q p I w ^ B W I _ y g P e ^ N I X c m E F Z e
L _ B l _ d o L a n C m E v M s _ m Y H X R R y P i u o X U ^ w S D I R o D D t Q r E X J K _ y l U y h J Z ^ K q O D a p Z i u Z d s q R j r L P K d
y Z a W a y q G c o C K E P B G J d ^ R j U M L y _ E _ K U R l o y o W H P U G Q u l _ j b q q i n o i k E h i l p N e l ^ A G B f _ G a l l e C O J a
o u r O d M e l g p u F S K Q s I u t j k _ O G E l J V L W x I U _ w g m x u L d _ K I J h v d f ^ b U S U T q o L L A H F h r p Z l i _ D o ^ t t M m u l
s L ^ P a k m k s L _ F U l q _ j o r ^ T e c P K a o c A R o w P H U ^ U s p B i u g M M b U T h K U X C q x D x l i ^ R G G y o r X A P T i n ^ O K p
_ Q A K E l a U O u t D d K o j W l q b S y m i u F I M g l i O C q G _ O s n D b X o P q G Z R h H g H m k t x o C P I L Y _ N a a J o S i c i a s ^ U
A e o m R m q y B d e d n M y f E v U J X l k z ^ X d g r s p w b r p ^ R N ^ P o L R e I R K U W I A X l u o Z q w l Y H H C B T e g ^ w s
c l e x M g t N C n a L w e l d h M e h H h i u X K D P I C L F d l W e u S l t x l j w a l g w m l j w j b e g e u r ^ q U e
l g m ^ J X S C O M l e k u f H l i a l s y m U f d M i t I C L F d l W e u S l t x l j w a l g w m l j w j b e g e u r ^ q U e
s J n a Z ^ R j x C ^ A k N y D Y U K G B g Q d N l D ^ t Q u H u x I S R n D x _ n B g e x v L n g m l P n K L E G C R ^ F B A l i i _ p ^ K n R y X I Q u b b U P
e Y R o y X C g K E Q y k N e ^ v H e c K p d e P U O A T l n g m l P n K L E G C R ^ F B A l i i _ p ^ K n R y X I Q u b b U P
g K J T f D _ G a C O D U y D b a w b q h J S B P y h Y r ^ m l W l n C s k v J Q i a E M e T O u y Y l h G ^ T s K C M c q r J C I S g S t d E r d c
W k e n P d ^ W R ^ U ^ P K X Y M q j j k B I F P Q H N ^ m l W l n C s k v J Q i a E M e T O u y Y l h G ^ T s K C M c q r J C I S g S t d E r d c
I K r x ^ Z N g k g a p Z _ d e ^ f p a o f C s ^ _ l C P A p O f G I S w M l r l e U ^ I G u c m P p k k n q B ^ d J I j s U _ k a F O u w F h X _ w d
p l ^ T g c j P t Q j h c g e B O f p a o f C s ^ _ l C P A p O f G I S w M l r l e U ^ I G u c m P p k k n q B ^ d J I j s U _ k a F O u w F h X _ w d
B U K X Q l a i l O p e L B x l e Q g s y p m l a T B s S B A l o _ h K U P U R F x ^ u n t U v u r D W ^ S N H m e v E r b b K q f F l y u d U B y G U R
q g E H q j l e P ^ _ ^ a y O a y E E E S w u c i D W X B s b K r Y M K W q b _ J T T a p v y j o _ w L J G M n ^ X u h g y n f x U S c n i f U y P K K M L p c
E F a G N Z x I L N I q K u u S y w G H y ^ P y p O M ^ X s b b K r Y M K W q b _ J T T a p v y j o _ w L J G M n ^ X u h g y n f x U S c n i f U y P K K M L p c
G g g k l M p b U H F D i p Q a u g e ^ a C y U y D e ^ U x ^ W L L O q m m I T S U k N U n R L J G M n ^ X u h g y n f x U S c n i f U y P K K M L p c
J a ^ J J I i Y a n M ^ F K y I W m O X u T I Z ^ T f P m e f f y B Z b q u c B i h Z ^ j p t q v e G ^ S a J S c r l a U n R d l E s O a I R k r l y p
c o f I m R l G i S P X S Q q ^ I u j H o i a d e q j l l k Q R y q y M i j O X o i i Q U B D u U U o M S R f G i k U Z n H ^ H s R R y j T M ^ T D _ a
D S c ^ U U I D I p l j c y ^ Z Z q I D M k h U j i W O M C t i U i n W y g o o H m a a m Z ^ B J i H p P R f i C P b W e w B W E a U Q D ^ U E S j e
R ^ q c y i k d W t a T o l l y I H P b y ^ X Z J s n l w h d g g F S p u s m Q t P Q _ Z U P a h u d C d C q b x W H o k H C P a j y F e X y a a O j R
```

# TROUBLESHOOTING WINDOWS PROBLEMS WITH POWERSHELL

Guy Leech ([@guyrleeche](https://twitter.com/guyrleeche))

PowerShell User Group Rhein-Neckar, February 2023



# GUY LEECH

- Independent consultant, developer, trainer, troubleshooter, comedian
- Citrix CTP, Microsoft MVP, VMware vExpert, Parallels VIPP
- [@guyleech](#)
- [guyleech.wordpress.com](#)
- [github.com/guyleech](#)
- [pastebin.com/u/guyleech](#)
- [www.youtube.com/c/GuyLeech42](#)
- [linkedin.com/in/guyleech/](#)
- Available for hire

# IN PERSON POWERSHELL WORKSHOPS



# WHY USE POWERSHELL FOR TROUBLESHOOTING?

- Consistency
- Speed
- Time of day (or night) – automation via scheduled tasks
- Lazy
  - Someone probably has done what you want already (but check script!)
  - Better things to do than do admin stuff
- Improve skills
- Remediate/fix too
- Easier to share
- Log analysis with regular expressions (regex)
- Less comeback



# WHAT SORT OF ISSUES?

- Processes
- Services
- Permissions
- Network
- Slow Logons
- Certificates
- Registry
- SQL
- Active Directory
- .....





# WMI/CIM

- Huge amount of available information (over 800 non performance classes by default)
- Tab completion of classes or list with Get-CimClass (PoSH v3+)
- Great way to get computer details and export to CSV for reference/analysis
- Some classes have methods which can be called, e.g. Win32\_UserProfile
- Filter in query, not afterwards if possible (speed, memory)
- Can take array of machines via -ComputerName
- No WMI calls in pwsh 7.x – use CIM
- Other name spaces, e.g. SCCM, Citrix
  - Get-CimInstance -Namespace Root -ClassName \_\_Namespace
  - Tab/control space completion so easy to explore

# SOME USEFUL WMI/CIM CLASSES

- Win32\_Process
  - Gives parent process details which Get-Process doesn't
  - Need to invoke GetOwner method to get owner via Invoke-CimMethod
  - If on multi-user OS, filter by SessionId if relevant
- Win32\_OperatingSystem
  - LastBootUpTime
- Win32\_LogonSession & Win32\_LoggedOnUser
  - Gives precise logon times for all logons since boot
- Win32\_ComputerSystem
- Win32\_Service
  - Executable including path which Get-Service doesn't
- Win32\_SystemDriver
- But don't use Win32\_Product as it isn't passive
  - Interrogate the registry
- Script to gather data via CIM for health checks, spot the difference, etc

# QUERYING EVENT LOGS

- There are over 300+ - how many have you been looking at?
- `Get-WinEvent -ListLog * | ? IsEnabled` (408 on my Win11 laptop)
- `Get-WinEvent -ListProvider *` (1255 on my Win11 laptop)
- Filter left for speed (hashtable, XPath or XML)
  - Hashtable can filter on event id, provider, log name, start & end times, level & more
  - `Get-WinEvent -FilterHashtable @{ LogName = 'Security' ; ID = 4688 ; StartTime = '17:00' ; EndTime = '17:20' }`
- Filter/select on Properties array rather than entire message
  - `Get-WinEvent -FilterHashtable @{ LogName = 'Security' ; ID = 4688 ; StartTime = '17:00' ; EndTime = '18:00' } | Where { $_.Properties[5].Value -match '\\cmd\\.exe' }`
- Much easier to visualise with Out-GridView than eventvwr
  - Can then filter in/out
  - Or save via Export-CSV
- Can be remoted so don't need to logon
- Script to bring all event logs together (was a one liner)



# GET-WINEVENT

- `Get-WinEvent -ListLog *terminalservices*`
- `Get-WinEvent -LogName Application`
- `Get-WinEvent -ProviderName 'Application Error' | Select-Object -First 10`
- Filtering
  - Don't filter afterwards (`Where-Object` or `.Where()`) if possible for speed
  - Uses XML, XPath or [Hash tables](#) (dictionaries)  
`Get-WinEvent -FilterHashtable @{ LogName = 'Security' ; ID = 4688 ; StartTime = '17:00' ; EndTime = '17:20' }`  
`Get-WinEvent -ListLog *|? RecordCount |%{ Get-WinEvent -EA Silent -FilterHashtable @{logname=$_.logname;starttime='16:29';endtime='16:31'}}|select *|sort TimeCreated|ogv`
  - Filter/select on Properties array rather than entire message  
`Get-WinEvent -FilterHashtable @{ LogName = 'Security' ; ID = 4688 ; StartTime = '17:00' ; EndTime = '18:00' }|Where { $_.Properties[5].Value -match '\\cmd\\.exe' }`
  - [Script to query all event logs in a given period](#)

# PROCESS AUDITING

- Enable creation & termination success auditing
  - GPO
  - Secpol.msc (Advanced Audit Policy Configuration->Detailed Tracking)
  - [auditpol.exe](#) (use GUIDs if non-English locale)
  - P/Invoke [AuditSetSystemPolicy](#)
  - Also [cmd line auditing](#) (potential security implications)
  - Increase Security event log size/make persistent
- Event ids 4688 and 4689 in Security event log
- Can troubleshoot as far back as the Security event log goes
  - Did something crash or exit prematurely (and if so what exit code)?
  - What launched that cmd.exe at logon?
  - Is something running a lot?
- [Script to show processes, durations, parents, command lines or summary](#)

# REMOTING

- Many cmdlets take –ComputerName and array of computers (comp1,comp2)
- Invoke-Command
- Winrm quickconfig
- Enter-PSSession
  - Similar to telnet/ssh access
  - Less resource intensive way to get access to troubled system
  - No GUI programs
  - Great for running SysInternals procmon headless, e.g. Windows 10
  - Accessing UNC requires extra configuration
- UNC access can be challenging within remoted commands/sessions

# POWERSHELL (PS) DRIVES

- Use standard cmdlets like Get-ChildItem, Remove-Item with various data sources
  - File system
  - Registry
  - Certificates
  - Variables
- Tab Completion
- Not all operations/properties implemented by all providers
- Additional PS providers
  - SQLPS (Use .NET SQL provider => one less dependency/pre-requisite)
  - VMware Datastores (can be slow, consider read-only /folder web interface)
  - Citrix CVAD datastore (generic interface to VMs & snapshots regardless of Hypervisor)

# ACTIVE DIRECTORY

- Install ActiveDirectory module or use built-in ADSI or WMI/CIM
- Great for bulk/complex queries and changes
- Report to csv/html and email via scheduled task, e.g. expired/expiring accounts
- Do big lookups and cache (hashtable) rather than lots of individual requests
- Be careful!
  - Use `-WhatIf/-Confirm (SupportsShouldProcess/ConfirmImpact/ShouldProcess)`
  - Backup/test in non-production
- GroupPolicy module
  - E.g. what GPOs have changed in the last 7 days?



# GUY'S TOP 10 TROUBLESHOOTING CMDLETS

1. Get-CIMInstance
2. Get-WinEvent
3. Enter-PSSession (etsn)
4. Out-GridView (ogv)
5. Export-CSV (epcsv)
6. Test-NetConnection (tnc)
7. Get-ADUser
8. Get-ChildItem (dir, gci, ls)
9. Get-Process (ps, gps)
10. Get-Command (gcm)

(In no particular order)



# BUBBLING UNDER

- AppVClient module
- Get-Counter – get any performance counter
- Get-AuthenticodeSignature – is file signed?
- Set-ACL – fix permissions, copy from a known good system
- Get-FileHash – are those two files the same? Is that download ok?
- \*-Service – start/stop services or change settings
- Stop-Computer/Restart-Computer – errors if anyone else logged on unless –force
- Invoke-WebRequest – is the web site working?

# EXAMPLE USAGE (1)

- Check port open (telnet.exe equivalent, ping can be too basic)  
`Test-NetConnection 192.168.0.4 -Port 443`
- Show expiring certificates  
`dir Cert:\LocalMachine\Root\? NotAfter -lt (Get-Date).AddDays( 90 ) |select subject,notafter`
- Show a specific process' CPU usage (no GUI)  
`Get-Date;ps -name tiworker|select -exp TotalProcessor*|select -exp TotalSeconds`
- Show overall CPU usage (no GUI)  
`Get-Counter -counter "\Processor(_Total)\% Processor Time"`
- Count registry keys (registry bloat issue giving slow logon)  
`dir "HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Group Policy" -Recurse|measure`
- Show all Citrix processes  
`ps |? Path -match 'Citrix'`

## EXAMPLE USAGE (2)

- When did that process/service start?  
`ps -name blah | Select id,starttime`
- Searching for files (for content)  
`dir searchfolder\*.xml -Force -Recurse|sls 'searchstring|regex'`
- What version are those files?  
`dir searchfolder\*.exe |select -expand VersionInfo`
- Show executable path & version info of a running process  
`ps -name process_name|gp -ea si|select -Expand VersionInfo`
- Show all McAfee services  
`Get-Service | ? DisplayName -match 'mcafee'`
- [Diagnose IIS/Web app issues via IIS logs](#)

# ONE LINERS

- Overrated as makes understanding difficult but can be useful – copy'n'paste

```
gc logfile|? { $_ -match '^(\\d|#Fields)' } | %{ $_ -replace '^#Fields: '
}|ConvertFrom-Csv -Del ' ' |select *,@{n='Duration';e={([int]$_.'time-
taken')}}|ogv

Get-WinEvent -ListLog * |?{ $_.RecordCount }|%{ Get-WinEvent -ea
SilentlyContinue -FilterH
@{logname=$_.logname;starttime='16:29:15';endtime='16:31:15'}}|select
*|sort TimeCreated|Out-GridView

dir "C:\\path" -force -Rec|?{ $_.PSIsContainer }|%{ if( ( Compare-Object
($acl = Get-Acl $_.FullName) (Get-Acl ($remote=$_.FullName -replace '^([A-
Z]):' , '\\machine2\\$1$')) -Property access)){ $acl | Set-Acl -Path
$remote}}

1..9|%{"{0,9} x 8 + $_ = $(8*($a=1..$_-join'')+$_)"-f$a}
```





# TIPS AND TRICKS #1

- Prefix/Suffix commands with Get-Date to record when ran for cross referencing
  - Get-Date; Test-NetConnection dodgyserver
  - Or set in Prompt() function in profile (\$profile)
- \$PSVersionTable
  - See what PoSH version you are running
- Ctrl r to search persistent history
- Tab complete & find Windows commands as well as PoSH ones
- Measure-Object
- Measure-Command
- Out-GridView (-PassThru)

## TIPS AND TRICKS #2

- Ctrl Backspace/Delete to delete whole word back/forward
- Ctrl Home/End to delete to start/end of line
- Ctrl z/y – undo (multiple)/redo
- Ctrl arrow – jump words
- Number conversions
  - `'{0:x}' -f 1234`
  - `0x4d2`
  - `[convert]::ToString( 1234 , 2 )`
- `scb/gcb` – pipe to/from clipboard
- `gcb | ConvertFrom-JSON`
- `(gcb).length` – how long is that string in the clipboard
- `(get-date) - (gcim Win32_OperatingSystem).LastBootUpTime` – what's the uptime?

# RUNNING POWERSHELL SCHEDULED TASKS

- Use to pre-empt problems, alert, fix, tidy, etc
- Use a service account with required permissions & group memberships
- Use a single machine, install all required modules, snapins, etc
- Create your own scheduled tasks folder for your tasks
- Make the script write a log file (Start-Transcript is good enough)
- Avoid clear text credentials in command lines – Secure String/Protect-CmsMessage
- Ensure service account has "Log on as a batch job" privilege
- How do you detect if scheduled task has failed?
- Test command line in cmd.exe, e.g. powershell.exe -file
- Parameter gotchas
  - \$true/\$false being treated as strings not bools
  - Arrays (comma separated) flattened into a single element
  - Quotes/spaces (or lack there of)
- Use PowerShell to create the scheduled task?
- Ensure script doesn't prompt

## OTHER USEFUL SCRIPTS

- Show registry keys modified in a given time period
- Show files changed since boot (e.g. Citrix PVS cache) or any time period
- Get chunk at file offset (e.g. tally procmon write to log file to contents)
- Digital clock/stopwatch/countdown timer (e.g. time & mark logon stages)
- Show/search loaded modules for processes (e.g. find hook dlls)
- Trim process working sets, set working set size limits (e.g. save memory)
- Get installed software (do not use Win32\_Product) (e.g. compare systems)
- Change CPU priorities (e.g. stop impact of runaway processes)
- Find/delete user profiles (e.g. infrastructure servers short of free disk space)



# HOW TO IMPROVE AT POWERSHELL

- Sit in PowerShell prompt not cmd
- Use it every working day
- Look at other people's scripts and understand them (how I started in 1980)
- Don't ignore automation opportunities but not everything is
- Understand and learn from errors – don't run away screaming
- Don't give up
- Bite the bullet – don't go back to the old ways of cmd, vbs, etc
- Everyone had to start somewhere
- Online training
- Communities – give and take (usually the other way round initially)
- Books





# FOOD FOR THOUGHT

- PowerShell is cross-platform – Mac & Linux too
- PowerShell is open source - <https://github.com/powershell>
- PowerShell v5.1 (latest/last Windows release) is EoL
  - PowerShell 7 is here (pwsh)
- Cmd batch scripting is painful & needs lots of exes for troubleshooting
- Powershell.exe can be slow to start compared with wscript.exe & cmd.exe
- Easy to create HTML/CSV and send SMTP emails – be proactive!



# RESOURCES

- <https://www.controlup.com/script-library/>
- <https://github.com>
- <https://gallery.technet.microsoft.com/scriptcenter> (deprecated)
- <https://devblogs.microsoft.com/scripting/>
- <https://4sysops.com>
- Your preferred search engine



EXIT( 0 )

- Thank you
- Live long and PowerShell