"POWERSHELL" IN "WINDOWS.."

by Francis Chao
fchao2@yahoo.com
Web location for this presentation:

• http://aztcs.org
• Click on “Meeting Notes”
"Powershell" has become an important feature of the "Windows.." operating system. In addition to being a feature-rich command line interpreter for power users and system administrators, it massively enhances any batch files the end-user creates and it allows access to many of the new features of "Windows 10" when they are initially available.
TOPICS

• "Powershell" Definitions
• "Powershell" History
• "Powershell" Window
• "Powershell" Cmdlets
• "Powershell" Scripts
• "Powershell" Allows Early Access to New Windows.. Features
TOPICS (continued)

Using "Powershell"

• Test a Motherboard's UUID
• Determine If "Secure Boot" Is Enabled Inside a Motherboard
• Start "Windows Defender Offline"
• "Powershell" to Repair The "Start Menu"
• "Powershell" to Enhance The Power of Batch Files
"POWERSHELL" DEFINITIONS

• According to Wikipedia: PowerShell .. is a task automation and configuration management framework from Microsoft, consisting of a command-line shell and associated scripting language... PowerShell provides full access to COM and WMI, enabling administrators to perform administrative tasks on both local and remote Windows systems..
"POWERSHELL" DEFINITIONS (continued)

- "PowerShell" is a command line interpreter that massively improves on what you can do at a command prompt inside "Windows XP.." and higher
"POWERSHELL" DEFINITIONS (continued)

• According to http://www.makeuseof.com/tag/command-prompt-vs-windows-powershell-whats-difference/ :
PowerShell has a lot of advanced features — like remote execution of tasks, background tasks, task automation, command piping, and more — that make it a better choice than the archaic Command Prompt when you have a lot of system administration and maintenance to do.
"POWERSHELL" HISTORY

• "Powershell 1.0" was introduced in November 2006 as an optional module for "Service Pack 2" of "Windows XP"
• October 2009: Available for "Windows Vista" as an optional download.
• Bundled module in "Windows 7", "Windows 8", "Windows 8.1" and "Windows 10"
"POWERSHELL" HISTORY (continued)

• August 2016: Alpha version of "Powershell" was released as free open-source software for Linux and MacOS ("OS X") (Windows.. version of "Powershell" remains closed-source.)
"POWERSHELL" WINDOW

• Many ways to start up a "Powershell" Window

• Even if you have an administrator account in a Windows computer, all of the default ways to start up a "Powershell" window result in a non-administrator version of "Powershell" that can cause lots of strange error messages
You can launch "Powershell" from inside any "Command Prompt" window in Windows 8 or higher by typing in or copying in with Ctrl + c:
powershell -Command "Start-Process PowerShell –Verb RunAs"
Then press the "Enter" key.
If you get a "User Account Control" box, click on "Yes".
"POWERSHELL" WINDOW (continued)

• You can launch "Powershell" from the inside the "Windows 10" "Start Menu":
  Locate the "Windows Powershell" folder.
  Click on it.
  Right-click on the "Powershell" menu selection.
  Click on "Run as Administrator".
"POWERSHELL" WINDOW (continued)

• If you accidently start "Powershell" as a non-administrator, you can elevate up to the administrator capability in "Powershell" by typing in or copying in

\[
\text{Start-Process PowerShell –Verb RunAs}
\]

and then hit the Enter key
"POWERSHELL" WINDOW (continued)

• The easiest way to start up a "Powershell" window as an "Administrator" is to create a shortcut as follows:
"POWERSHELL" WINDOW (continued)

- Step 1: Use the RIGHT mouse button to click on the Windows "Desktop":
"POWERSHELL" WINDOW (continued)

• Step 2: Click on "New" in the pop-up context menu:
"POWERSHELL" WINDOW (continued)

• Step 3:
  Click on "Shortcut":


"POWERSHELL" WINDOW (continued)

- Step 4:
  A "Create Shortcut" dialog box will be displayed:
What item would you like to create a shortcut for?

This wizard helps you to create shortcuts to local or network programs, files, folders, computers, or Internet addresses.

Type the location of the item:

| Type the location of the item: | Browse... |

Click Next to continue.
"POWERSHELL" WINDOW (continued)

• Step 5: In the "Location.." field, type or copy in
  C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
Type the location of the item:

C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
"POWERSHELL" WINDOW (continued)

• Step 6:
  Click on "Next":
Create Shortcut

What item would you like to create a shortcut for?

This wizard helps you to create shortcuts to local or network programs, files, folders, computers, or Internet addresses.

Type the location of the item:
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

Click Next to continue.

Next  Cancel
"POWERSHELL" WINDOW (continued)

• Step 7:
  Click on "Finish":

Create Shortcut

What would you like to name the shortcut?

Type a name for this shortcut:

powershell.exe

Click Finish to create the shortcut.
"POWERSHELL" WINDOW (continued)

- Step 8: Locate the "powershell.exe" shortcut on the Windows "Desktop":
"POWERSHELL" WINDOW (continued)

• Step 9: Use the RIGHT mouse button to click on the "powershell.exe" shortcut:
"POWERSHELL" WINDOW (continued)

• Step 10:
  Click on "Properties" in the pop-up context menu:
"POWERSHELL" WINDOW (continued)

• Step 11:
  Click on "Advanced": 

"POWERSHELL" WINDOW (continued)

• Step 12:
Place a checkmark for "Run as administrator:"
Choose the advanced properties you want for this shortcut.

- Run as administrator
  This option allows you to run this shortcut as an administrator, while protecting your computer from unauthorized activity.
- Run in separate memory space

[OK] [Cancel]
"POWERSHELL" WINDOW (continued)

• Step 13:
  Click on "OK":


Choose the advanced properties you want for this shortcut.

- Run as administrator
  This option allows you to run this shortcut as an administrator, while protecting your computer from unauthorized activity.
- Run in separate memory space

[OK] [Cancel]
"POWERSHELL" WINDOW (continued)

• Step 14:
  Click on "Apply":
"POWERSHELL" WINDOW (continued)

• Step 15:
  Click on "OK":


According to Wikipedia, Cmdlets are specialized commands in the PowerShell environment that implement specific functions. These are the native commands in the PowerShell stack. Cmdlets follow a Verb-Noun naming pattern, such as Get-Childitem, helping to make them self-descriptive.
Administrator: powershell.exe - Shortcut

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> get-childitem
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> get-childitem

Directory: C:\Windows\system32

<table>
<thead>
<tr>
<th>Mode</th>
<th>LastWriteTime</th>
<th>Length</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>d----</td>
<td>7/16/2016 7:14 AM</td>
<td>0409</td>
<td>0409</td>
</tr>
<tr>
<td>d----</td>
<td>7/16/2016 4:47 AM</td>
<td></td>
<td>AdvancedInstallers</td>
</tr>
<tr>
<td>d----</td>
<td>7/16/2016 4:47 AM</td>
<td></td>
<td>AppLocker</td>
</tr>
<tr>
<td>d----</td>
<td>10/27/2016 4:00 PM</td>
<td></td>
<td>appraiser</td>
</tr>
<tr>
<td>d----</td>
<td>10/28/2016 6:15 AM</td>
<td></td>
<td>ar-SA</td>
</tr>
<tr>
<td>d----</td>
<td>10/28/2016 6:15 AM</td>
<td></td>
<td>bg-BG</td>
</tr>
<tr>
<td>d----</td>
<td>12/13/2016 5:51 PM</td>
<td></td>
<td>Boot</td>
</tr>
</tbody>
</table>
The Get-ChildItem cmdlet in Powershell is similar to dir in the conventional "Windows.." command prompt or at the command line in MS-DOS.
"POWERSHELL" SCRIPTS

- "Powershell" scripts are text files that have a .ps1 file extension.
- For security reasons, to keep malware from running malicious "Powershell" scripts, "Powershell" scripts do not run by default until the computer user gives them permission to run with `Set-ExecutionPolicy RemoteSigned` as mentioned at [http://ss64.com/ps/syntax-run.html](http://ss64.com/ps/syntax-run.html)
"POWERSHELL" SCRIPTS

To get a Powershell script to pause at the end, put in Write-Host "Press any key to continue ..."

$x = $host.UI.RawUI.ReadKey("NoEcho,IncludeKeyDown")

Write-Host
Write-Host "A"
Write-Host "B"
Write-Host "C"

at the end of the script as mentioned at

"POWERSHELL" SCRIPTS (continued)

• Some useful "canned" Powershell scripts can be found at http://www.robvanderwoude.com/powershellexamples.php
"POWERSHELL" SCRIPTS (continued)

• For most computer end-users, the best way to run a "Powershell" script is to use a .bat file with the following code in it:
"POWERSHELL" SCRIPTS (continued)

SET ThisScriptsDirectory=%~dp0
SET PowerShellScriptPath=%ThisScriptsDirectory%script1.ps1
PowerShell -NoProfile -ExecutionPolicy Bypass -Command "& '%PowerShellScriptPath%'";
"POWERSHELL" SCRIPTS (continued)

• `script.ps1` should be changed to the actual name of the script that you wish to run
"POWERSHELL" ALLOWS EARLY ACCESS TO NEW WINDOWS.. FEATURES

• When Microsoft is about to introduce a new feature in "Windows..", they often make it available first as a Powershell command

• Then one or two years later, the new feature finally shows up in the graphical user interface (GUI) screens in the "Windows.." operating system
"POWERSHELL" ALLOWS EARLY ACCESS TO NEW WINDOWS... FEATURES (continued)

• For example, the ability to support "Secure Boot" in "Hyper-V" virtual machines running Linux in a "Windows 8.1 Pro" host, was first introduced in January 2015.
"POWERSHELL" ALLOWS EARLY ACCESS TO NEW WINDOWS.. FEATURES (continued)

• Back in January 2015 this new feature could only be enabled by means of the Powershell command of Set-VMFirmware "vmname" - SecureBootTemplate MicrosoftUEFICertificateAuthority where "vmname" is the assigned name of the virtual machine as mentioned in http://windowsitpro.com/hyper-v/secure-boot-linux-virtual-machine-hyper-v
"POWERSHELL" ALLOWS EARLY ACCESS TO NEW WINDOWS.. FEATURES (continued)

• Then one year later in March 2016, the new feature could be enabled in the graphical user interface (GUI) screens of the "Hyper-V" window in the "Windows 10 Pro" operating system as described at https://blogs.technet.microsoft.com/dubai/sec/2016/03/29/secure-boot-on-virtual-machines/:
Use Secure Boot to help prevent unauthorized code from running at boot time (recommended).

- Enable Secure Boot

Template:
- Microsoft Windows
- Microsoft Windows
- Microsoft UEFI Certificate Authority

A Trusted Platform Module (TPM) is a special purpose microprocessor which provides cryptographic services to a compute platform.
Settings for Win10

Hardware
- Add Hardware
- Firmware
  - Boot from Network Adapter
- Security
  - Secure Boot enabled
- Memory
  - 1024 MB
- Processor
  - 1 Virtual processor
- SCSI Controller
- Network Adapter
  - private

Security

Secure Boot
Use Secure Boot to help prevent unauthorized code from running at boot time (recommended).

- [ ] Enable Secure Boot

Template:
- Microsoft Windows
- Microsoft Windows
- Microsoft UEFI Certificate Authority

A Trusted Platform Module (TPM) is a special purpose microprocessor which provides cryptographic services to a compute platform.
WINdows usERS Computer user group

Irvine Water District Facilities
Sand Canyon Meeting Room
15600 Sand Canyon Ave, Irvine, CA 92618

10:00 am to 12:30 pm

If you are interested in learning more about Windows, a particular program, or even an exchange of ideas or information regarding the computer industry, WINNERS - WINdows usERS is for you! WINNERS is a great place to see the latest software and also learn how to use it. We provide an easy (and fun) atmosphere for you to see new and interesting programs or hardware. Each month we have demonstrations of how to work with different programs, or hardware. We also have a discussion of computer and software problems (Random Access), and how to solve them. Computer tips are given out each month. You don't have to join right away; come in and see if you like us. We are in Orange County, California, right near the San Diego (405) Freeway.

Membership is $20 annually for individuals with $5 for each additional family member.

"User groups" represent the spirit of the frontier, a community getting together to do things that no individual ought.

Firefox automatically sends some data to Mozilla so that we can improve your experience.
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD

- You can use "Powershell" to test the UUID of the motherboard in "Windows 10":
  Start an administrative "Powershell" window.
  Then type in or copy in
  `get-wmiobject Win32_ComputerSystemProduct | Select-Object -ExpandProperty UUID`
get-wmiobject Win32_ComputerSystemProduct | Select-Object -ExpandProperty UUID
Administrator: powershell.exe

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation

PS C:\WINDOWS\system32> get-wmiobject Win32A24D56-E4E7-DAC1-B430-C0E77D4FF345
PS C:\WINDOWS\system32>
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD (continued)

• If you upgrade to or "activate" Windows 10, Microsoft stores the UUID of the motherboard of the computer up in their activation servers in their server farm system. Microsoft tests for this UUID whenever you boot up "Windows 10" and at other times when "Windows 10" is running. If your motherboard fails to have this 128-bit UUID stored inside firmware, your computer will lose it's "activation".
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD (continued)

• For a "Windows 8.1" computer, if your motherboard's UUID gets zeroed out because of a firmware failure in the motherboard, your "activation" will fail, as stated at https://mikebeach.org/2013/01/22/get-smbios-uuid-using-wmic/
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD (continued)

• Reference for how activation works in "Windows 10":  
  http://aztcs.org/meeting_notes/winhardsg/win10/activation/win10-activation.pdf

• Reference for this "Powershell" command:  
When "Windows 10" was first released, the previously-mentioned "Powershell" command was the only reliable way to check the UUID of the motherboard of a "Windows 10" computer. However, we now have two additional ways to read the UUID of the motherboard: We can now run a wmic command in a command prompt and we can now run a .vbs visual basic script:
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD (continued)

- To get the motherboard UUID from a regular or elevated command prompt, type in `wmic csproduct get uuid` OR type in `wmic path win32_computersystemproduct get uuid`
"POWERSHELL" TO TEST THE UUID OF THE MOTHERBOARD (continued)

• To create a .vbs visual basic script to get the UUID of a motherboard in "Windows 10", see
  http://www.out-web.net/?p=241
"POWERSHELL" TO SEE IF "SECURE BOOT" IS ENABLED IN THE MOTHERBOARD

• You can use "Powershell" to see if "Secure Boot" is enabled in the motherboard. Then type in or copy in confirm-securebootuefi
"POWERSHELL" TO SEE IF "SECURE BOOT" IS ENABLED IN THE MOTHERBOARD

• If your computer has a "Secure Boot" module (inside a UEFI) that is "enabled", you will get a response of True:
Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> confirm-securebootuefi
Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> confirm-securebootuefi
True
PS C:\Windows\system32>
"POWERSHELL" TO SEE IF "SECURE BOOT" IS ENABLED IN THE MOTHERBOARD

• If your computer does has a "Secure Boot" module in the UEFI but the "Secure Boot" module is not enabled, you will get a response of False
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32> powershell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> confirm-securebootuefi
PS C:\Windows\system32> confirm-securebootuefi
False
PS C:\Windows\system32> .
"POWERSHELL" TO SEE IF "SECURE BOOT" IS ENABLED IN THE MOTHERBOARD

• If your computer does not have a "Secure Boot" module in the UEFI or the "Compatibility Support Model" ("CSM") of the UEFI is enabled or it has a BIOS which would never have a "Secure Boot" module, you will get a response that "variable is currently undefined": 
confirm-securebootuefi : Variable is currently undefined: 0xC0000100
At line:1 char:1
+ confirm-securebootuefi
  + ~~~~~~~~~~~~~~~~~~~~
  + CategoryInfo : ResourceUnavailable: (Microsoft.SecureBootUefiCommand:ConfirmSecureBootUefiCommand) [ConfirmSecureBootUefiCommand]
  + FullyQualifiedErrorId : GetFwVarFailed,Microsoft.SecureBoot.Commands.ConfirmSecureBootUefiCommand
PS C:\WINDOWS\system32> confirm-securebootuefi
confirm-securebootuefi : Variable is currently undefined: 0xC0000100
At line:1 char:1
+ confirm-securebootuefi
+ ~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo : ResourceUnavailable: (Microsoft.Secur.
onfirm-SecureBootUEFI], StatusException
"POWERSHELL" TO SEE IF "SECURE BOOT" IS ENABLED IN THE MOTHERBOARD

• When "Windows 8" was first released, you could not determine if "Secure Boot" was enabled in your motherboard without this "Powershell" command. However, you can now run msinfo32 to see if "Secure Boot" is enabled:
Run

Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: 

OK  Cancel  Browse...
Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: msinfo32
<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS Name</td>
<td>Microsoft Windows 10 Pro</td>
</tr>
<tr>
<td>Version</td>
<td>10.0.14393 Build 14393</td>
</tr>
<tr>
<td>Other OS Description</td>
<td>Not Available</td>
</tr>
<tr>
<td>OS Manufacturer</td>
<td>Microsoft Corporation</td>
</tr>
<tr>
<td>System Name</td>
<td>DESKTOP-32LSCSJ</td>
</tr>
<tr>
<td>System Manufacturer</td>
<td>MSI</td>
</tr>
<tr>
<td>System Model</td>
<td>MS-7971</td>
</tr>
<tr>
<td>System Type</td>
<td>x64-based PC</td>
</tr>
<tr>
<td>System SKU</td>
<td>Default string</td>
</tr>
<tr>
<td>Processor</td>
<td>Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz, 3.408 Mhz, 4 Core...</td>
</tr>
<tr>
<td>BIOS Version/Date</td>
<td>American Megathreads Inc. A.A0, 7/27/2016</td>
</tr>
<tr>
<td>SMBIOS Version</td>
<td>3.0</td>
</tr>
<tr>
<td>Embedded Controller V...</td>
<td>255.255</td>
</tr>
<tr>
<td>BIOS Mode</td>
<td>UEFI</td>
</tr>
<tr>
<td>BaseBoard Manufacturer</td>
<td>MSI</td>
</tr>
<tr>
<td>BaseBoard Model</td>
<td>Not Available</td>
</tr>
<tr>
<td>BaseBoard Name</td>
<td>Base Board</td>
</tr>
<tr>
<td>Platform Role</td>
<td>Desktop</td>
</tr>
<tr>
<td>Secure Boot State</td>
<td>On</td>
</tr>
<tr>
<td>PCR7 Configuration</td>
<td>Binding Not Possible</td>
</tr>
<tr>
<td>Windows Directory</td>
<td>C:\Windows</td>
</tr>
<tr>
<td>System Directory</td>
<td>C:\Windows\System32</td>
</tr>
<tr>
<td>Boot Device</td>
<td>\Device\HarddiskVolume2</td>
</tr>
<tr>
<td>Locale</td>
<td>United States</td>
</tr>
<tr>
<td>Hardware Abstraction L...</td>
<td>Version = &quot;10.0.14333.206&quot;</td>
</tr>
<tr>
<td>User Name</td>
<td>DESKTOP-32LSCSJ\Restuser</td>
</tr>
<tr>
<td>Time Zone</td>
<td>Pacific Standard Time</td>
</tr>
<tr>
<td>Installed Physical Mem</td>
<td>32.0 GB</td>
</tr>
<tr>
<td>BIOS Mode</td>
<td>UEFI</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>BaseBoard Manufacturer</td>
<td>MSI</td>
</tr>
<tr>
<td>BaseBoard Model</td>
<td>Not Available</td>
</tr>
<tr>
<td>BaseBoard Name</td>
<td>Base Board</td>
</tr>
<tr>
<td>Platform Role</td>
<td>Desktop</td>
</tr>
<tr>
<td>Secure Boot State</td>
<td>On</td>
</tr>
<tr>
<td>PCR7 Configuration</td>
<td>Binding Not Possible</td>
</tr>
<tr>
<td>Windows Directory</td>
<td>C:\Windows</td>
</tr>
<tr>
<td>System Directory</td>
<td>C:\Windows\system32</td>
</tr>
<tr>
<td>Boot Device</td>
<td>\Device\HarddiskVolume2</td>
</tr>
</tbody>
</table>
"POWERSHELL" TO START "WINDOWS DEFENDER OFFLINE"

• You can use "Powershell" to start "Windows Defender Offline Mode" to remove malware infections: Launch PowerShell as Administrator, and then run the following command:
  Start-MpWDOSScan
"POWERSHELL" TO START "WINDOWS DEFENDER OFFLINE"

• You can also create a shortcut that runs "Powershell" and then starts "Windows Defender Offline". See
  https://www.tenforums.com/tutorials/42603-windows-defender-offline-scan-shortcut-create-windows-10-a.html
Administrator: Windows PowerShell

Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> confirm-securebootuefi
Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> confirm-securebootuefi
True
PS C:\Windows\system32>
Administrator: powershell.exe
PS C:\WINDOWS\system32> confirm-securebootuefi
confirm-securebootuefi : Variable is currently undefined: 0xC0000100
At line:1 char:1
+ confirm-securebootuefi
+ ~~~~~~~~~~~~~~~~~~~
   + CategoryInfo : ResourceUnavailable: (Microsoft.SecureBootUefiCommand:ConfirmSecureBootUefiCommand) [Confirm-SecureBootUEFI], StatusException
   + FullyQualifiedErrorId : GetFwVarFailed,Microsoft.SecureBoot.commands.ConfirmSecureBootUefiCommand
"POWERSHELL" TO ENHANCE THE POWER OF BATCH FILES

• You can use "Powershell" to enhance batch files.
"POWERSHELL" TO ENHANCE THE POWER OF BATCH FILES

• You can use "Powershell" to make batch files that are more powerful than in the past.