

Computer Systems

8. Windows scripting



Review

- Computers, information, number representation, code writing, architecture, file systems
- Base commands, foreground and background processes
- I/O redirection, filters, regular expressions
- Variables, command substitution, arithmetical, logical expressions
- Script control structures, sed, awk
- Network services

What comes today?

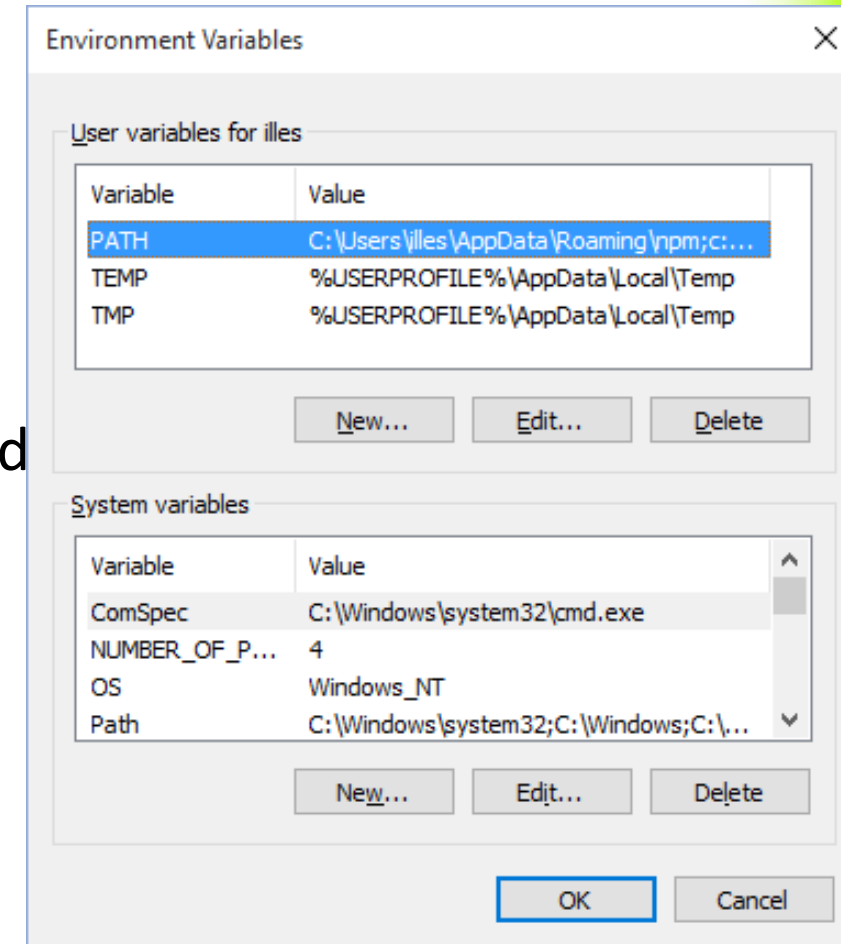
- Windows client
 - Services
- The automation of operations
 - Batch commands
 - Batch incompleteness
 - Windows Script Host
- PowerShell

Windows operating system

- The past: IBM PC DOS – Disk Operating System
 - 86-DOS – CP/M clone, Seattle Computer Products
 - Microsoft bought it in 1981.
 - It was renamed to : MS-DOS , 1981
- 1984 – Microsoft Windows appeared as the graphical extension of MS-DOS.
- 1993 – 32 bit system, Windows NT
- 1995 – Windows 95, Windows NT 4.0
- Today: 2015 – Windows 10

Windows user interface

- Graphical interface, to start a command – click to an icon
 - Start menupoint, task bar, desktop
- There is a command window (cmd)
 - Each installed program can be started from command line as well!
- Path environment variable consists of 2 parts!
 - User path + system path



Is there a possibility to create scripts?

- Yes!
- Primarily it is for making the administration easier.
 - The like as the possibilities in UNIX, Linux!
- There are several script types in MS Windows!
 - Batch program (bases)
 - Windows Script(ing) Host (VB Script or JScript based)
 - PowerShell

Bases of batch programs

- It appeared as a part of MS DOS system!
- Main task: To collect commands and start them together with the help of one ,batch' command.
- It is used (but rarely) in these days as well!
- Text commands
- Filename extension: .bat
- Remark: rem
- echo instruction
- call other.bat

Batch variables, parameters

- It is not case sensitive!
- Variable definition: set a=5
- To write out each variable: set
- Value of variable: %name%, etc: %a%
- path command, %path% variable
- prompt instruction
- %1, %9 batch parameters
- %0 the name of the batch program
- shift command, shifts the parameters to the left

Batch control structures

- Label definition: :label1
- Jumping to a label: goto label1
- Branch: if [not] condition instruction
 - if errorlevel 5 goto labelfive
 - True, if the return value of the previous command is greater or equal to 5!!
 - if %a%=="5" goto 5
 - if exist filename goto six
- Cycle: for %%variable in (list) do (instruction)

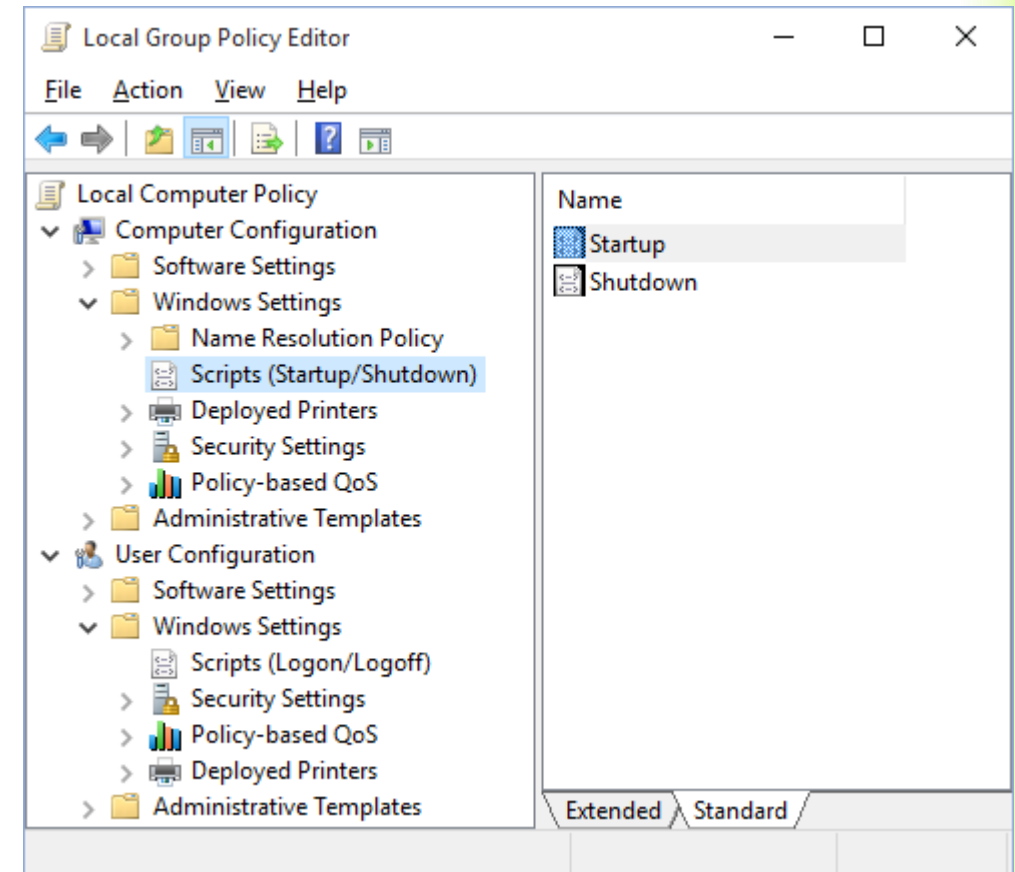
Batch cycle example

- It is similar to shell script for cycle
- In XP /f switch

```
rem for cycle example
rem do not write out commands
@echo off
rem you should use %%name form!!!
for %%i in (apple pear) do (
    echo %%i
)
rem read the first words of each line (in a file)
for /f %%j in (apple.txt) do (
    echo %%j
)
```

autoexec.bat vs. GPEDIT.MSC

- It's function: it is the collection of commands (batch) to be executed automatically at login!
- Today it is replaced by the possibility of creating group policy
 - Windows startup/shutdown
 - User startup/shutdown

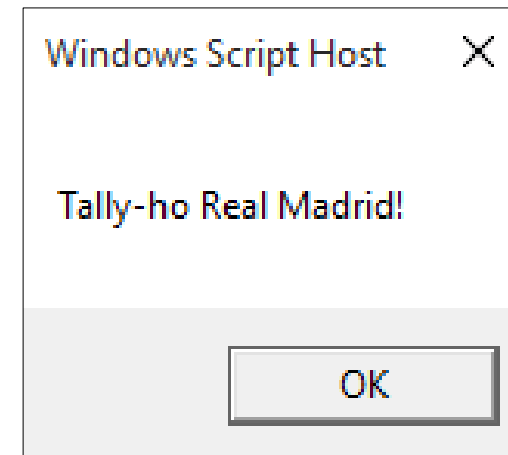
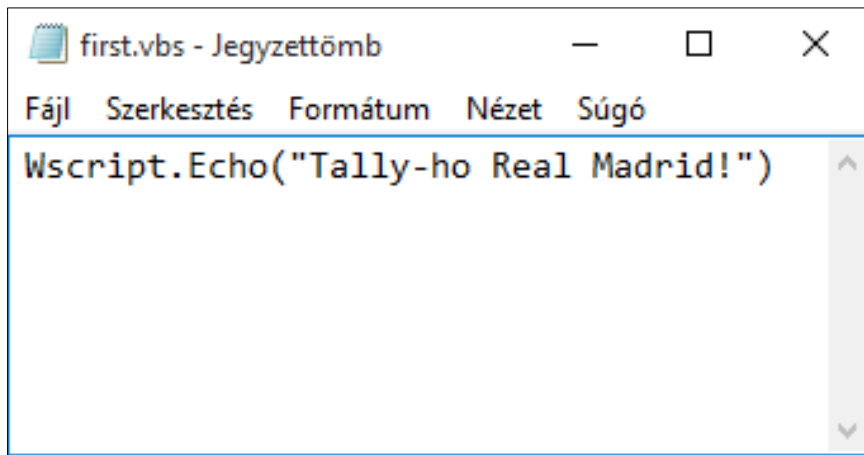


Windows Script Host - VBScript

- It appeared in Windows 2000 server – it is a new script language
 - It is not really new, it had appeared in MS Office 97 as a new macro language!
 - General script environment (Visual Basic Script, JScript though it is rarely used)
 - The macro language of the most modern Office is the same!
- .vbs extension (.js can be too, Jscript)
- Description: MSDN documentation
 - [https://msdn.microsoft.com/en-us/library/9bbdkx3k\(v=vs.84\).aspx](https://msdn.microsoft.com/en-us/library/9bbdkx3k(v=vs.84).aspx)

WSH example

- One should create a command file with .vbs or .js file extension!
- WSH object model, Wscript is the root object!
 - Echo method
- Execution: From command line or with mouse click



What is the next after WSH?

- While WSH is still used in these days, a new script tool appeared with .NET Framework!
- .NET Framework 2.0, 2005
- PowerShell

PowerShell

- It is the new generation of Microsoft script languages
 - Batch, VBS, WSH successor
- It is free of charge, it had to be installed under XP, Vista now it is already part of WS 2008, 2012, Win7, Win8, Win10
- You can download from: <http://www.microsoft.com/>. It is free!
- Current version: 3.0 (Windows 8), 4.0 (Win 8.1), 5.0 (Win10)
 - In Windows 7 there is Powershell 2.0 but you can upgrade it!
 - You can use Windows Management Framework 4 in Win7 too!
 - Get-PSSnapin – Gives back the PS version!

Powershell accessibility - Install

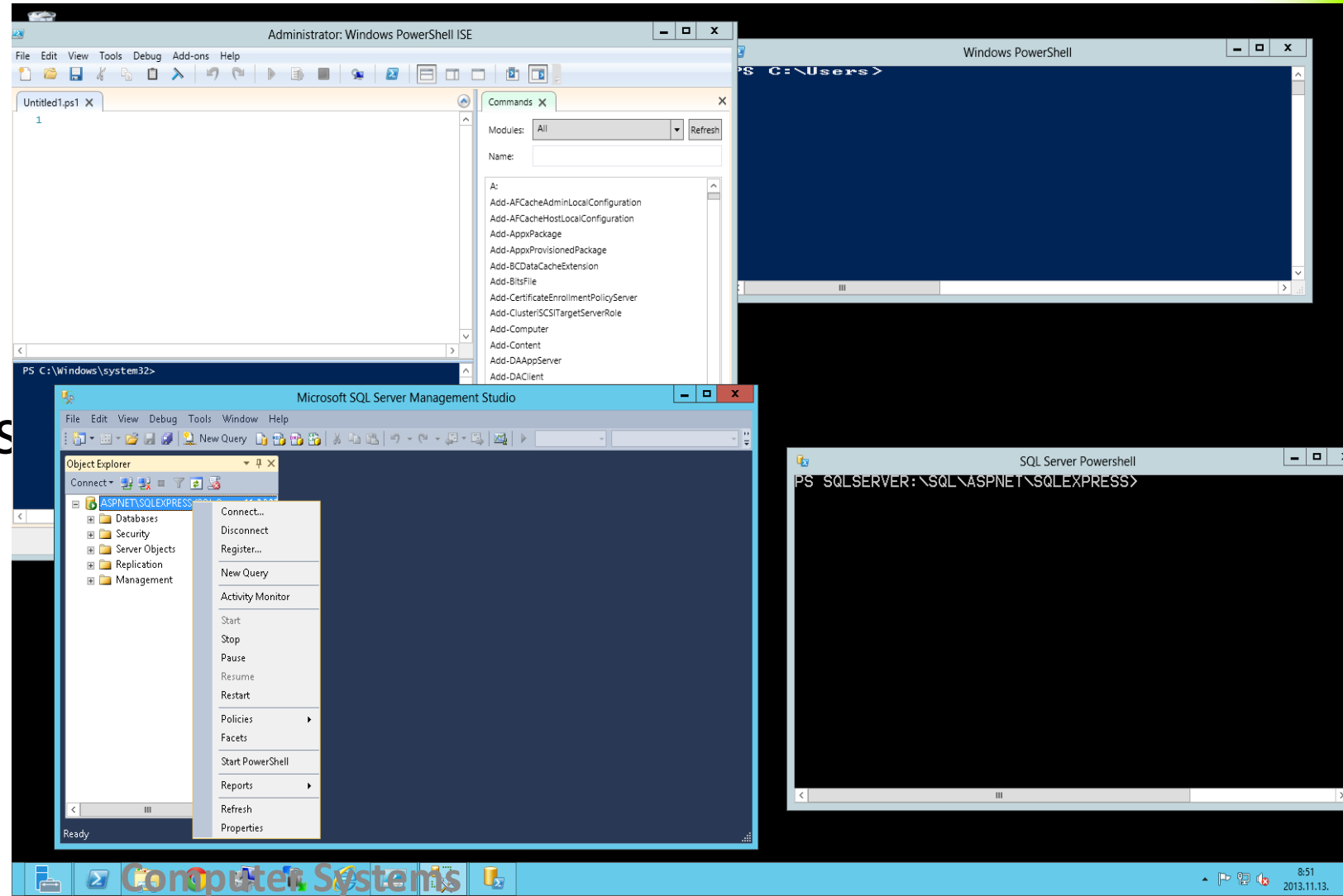
- .NET Framework 2.0 szükséges
- Start- Programs- Windows Powershell program group
- You have to install it under Windows XP!
 - PowerShell 1.0 Documentation Pack
- Documentation: online help
- Command appears in a character window similar to cmd (command.com) or on a graphical environment (ISE).

What it is good for PowerShell?

- For the same as shell script!
- Primarily for management purposes
 - First official Ms course: Course 6434A (2008)
 - Automating Windows Server® 2008 Administration with Windows PowerShell
 - General script programming environment under Windows operation system.

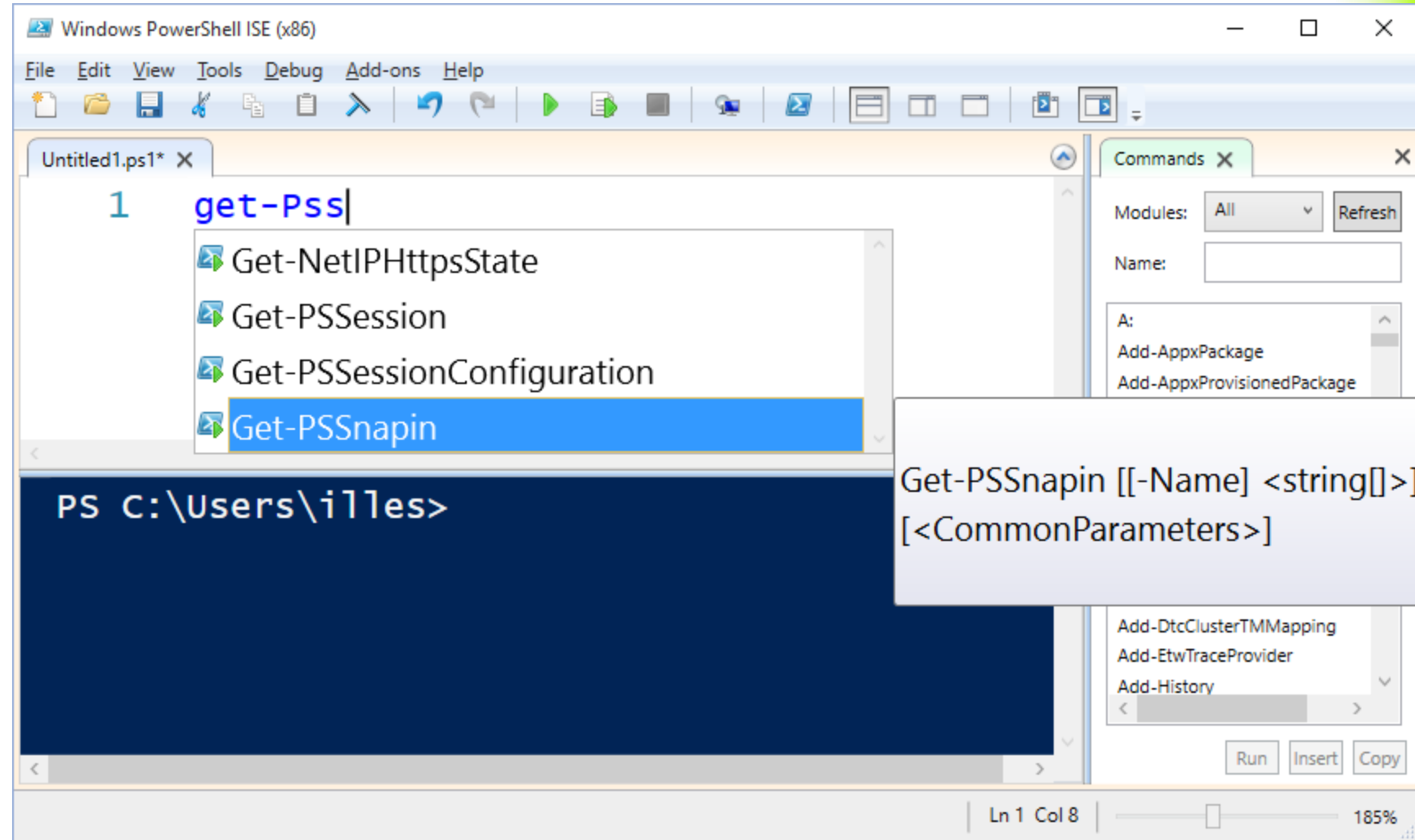
Powershell „versions”

- Core, the part of each Windows operating systems
- Services (MS SQL, Web, Exchange) gives extensions (module) to the base system!



PowerShell (core) today

- Ordinary graphical possibilities (File, Edit, View)
- Tools – working window settings: font, color.
- Debug – Usual debugging possibilities!



PowerShell Debug

- F9- Toggle Breakpoint
 - Usual step over, etc.
 - Only in ISE (F5)!!!
 - Starting from the command window there is no effect of the breakpoints!
- You may write out the script variables!
- Special variables:
 - The inner pipeline variables of the debugger!
 - \$Args
 - \$MyInvocation
 - \$Input
 - \$_
 - \$PSBoundParameters

The screenshot shows the Windows PowerShell ISE (x86) interface. The script editor displays a PowerShell script with the following content:

```
1 "hajra"
2 for($i=1;$i -lt 10;$i++)
3 {
4     $i.ToString()+"Fut a powershell"
5 }
6 $max=0
7 $nev="almás"
8 dir|ForEach-Object -Process { if ($_.Length -gt $max)
9 { $max=$_.Length
10  $nev=$_.FullName
```

A breakpoint is set at line 8, indicated by a small square icon. The debug console at the bottom shows the following output:

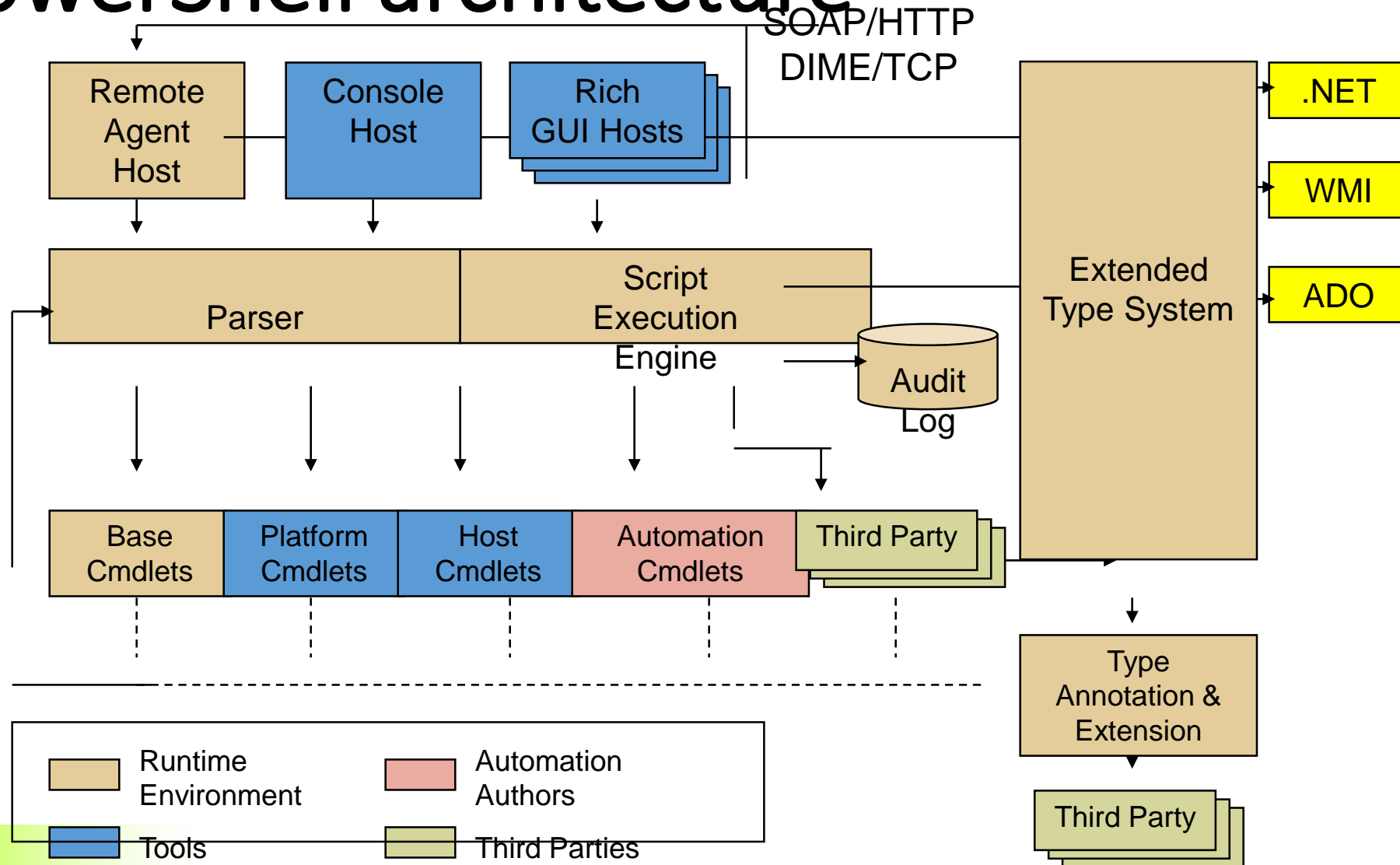
```
[DBG]: PS C:\Users\illes>> $max
0

[DBG]: PS C:\Users\illes>> $i
10

[DBG]: PS C:\Users\illes>>
```

The status bar at the bottom indicates "Completed", "Ln 45 Col 28", and "185%".

PowerShell architecture



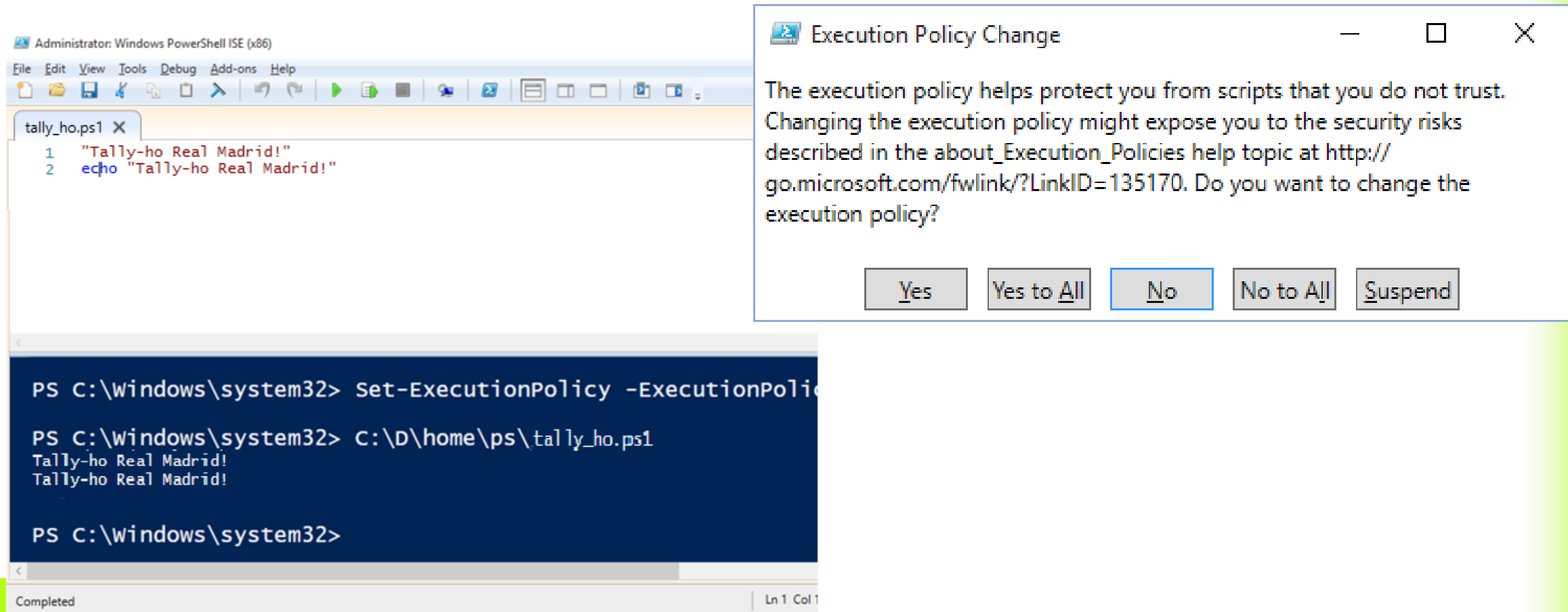
Modified:2018. 11. 03.

Safety script execution

- Get-ExecutionPolicy
- Set-ExecutionPolicy –ExecutionPolicy UnRestricted
 - Default: Restricted – execution is not permitted!
 - Possible policy values: Allsigned, Remotesigned, Bypass
 - At Unrestricted it will ask at downloaded scripts, at Bypassnál it will not ask!
 - -scope process or currentuser or localmachine
 - Remotesigned: at scripts downloaded from the net it will execute them only if they are signed by a trustable partner.
- Important! PowerShell starting: Run As Administrator!

PS script execution

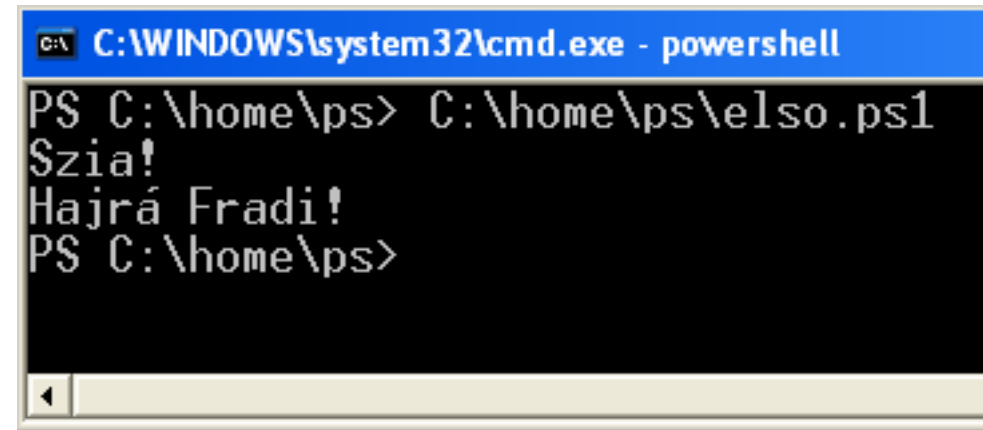
- Filename extension: ps1



PS script execution

- Command execution with typeing full path
- If there is a space in the path then write & symbol before the command and write it among "".
 - &"c:\apple tree\jonatan.ps1"

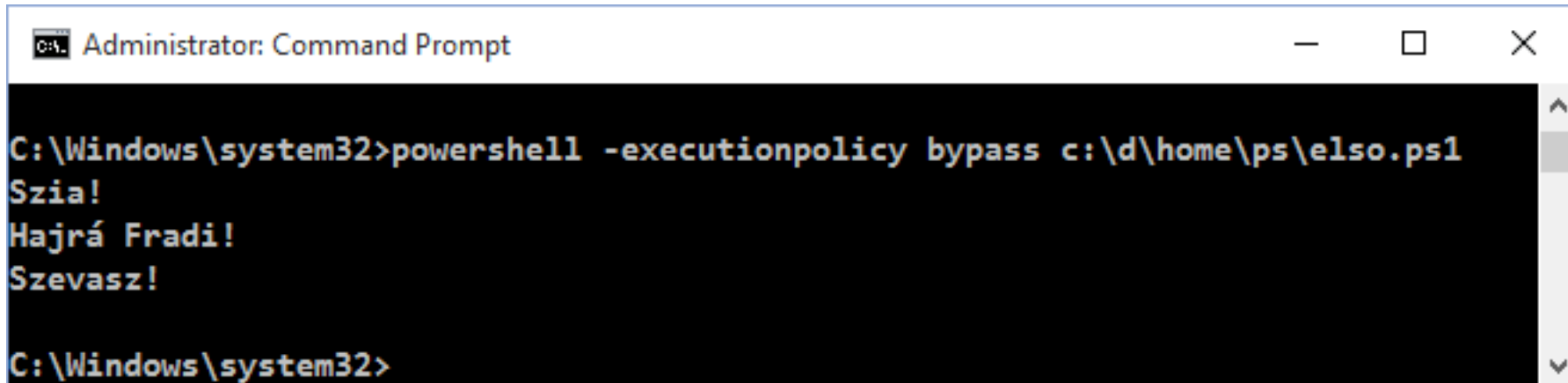
```
# Remark  
echo Szia!  
#  
Write-Host "Hajrá Fradi!"
```



```
C:\> C:\WINDOWS\system32\cmd.exe - powershell  
PS C:\home\ps> C:\home\ps\elso.ps1  
Szia!  
Hajrá Fradi!  
PS C:\home\ps>
```


PS script execution from command line

- You should give the script as a parameter!
- You should give the executionpolicy parameter! (if it is not set!)



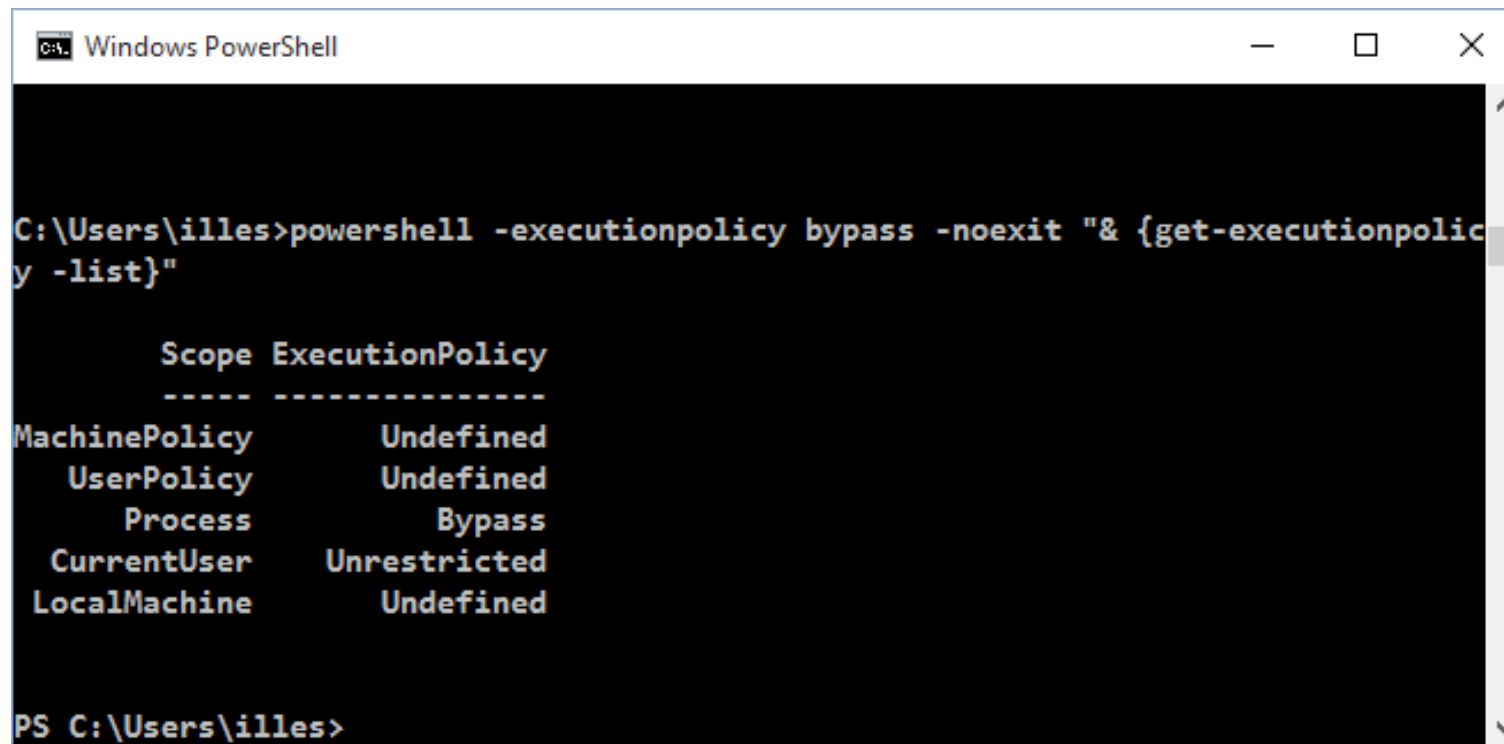
```
Administrator: Command Prompt

C:\Windows\system32>powershell -executionpolicy bypass c:\d\home\ps\elso.ps1
Szia!
Hajrá Fradi!
Szevasz!

C:\Windows\system32>
```

PS command execution

- PS parameter: "& {command}"
- -noexit it does not quit from PS



```
CA: Windows PowerShell
C:\Users\illes>powershell -executionpolicy bypass -noexit "& {get-executionpolicy -list}"

Scope ExecutionPolicy
-----
MachinePolicy Undefined
UserPolicy Undefined
Process Bypass
CurrentUser Unrestricted
LocalMachine Undefined

PS C:\Users\illes>
```

PowerShell commandline

- Writing a constant the interpreter tries to fit to a type!
- Default operation is write-host (if we do not say anything else)

```
PS C:\Users\illes> realmadrid
The term ,realmadrid is not recognized as a cmd in. At
line:1 char:10 + realmadrid <<<<
PS C:\Users\illes> „realmadrid"
realmadrid
PS C:\Users\illes> 5
5
PS C:\Users\illes> 'F'
F
PS C:\Users\illes> F
The term 'F' is not recognized as a cmdlet, At line:1char:1
+ F <<<<
```

Forms of PS commands

- Two main families (initially, nowadays more 😊)
 - Get-command family, Set-command family
- PS command : commandlets- cmdlets
 - Built in commands + extensions.
 - „Traditional commands” (echo) may be used!
- Tab key completes the command
- Remark: #
 - Several lines long remark: <##>
 - If at the beginning of the script there are one of the followings blocks:
.Description, .Syntax, .Synopsis, .Examples then the get-help is based on it!
- It is not case sensitive

PowerShell Help

- The command of help: Get-Help
- Only the syntax of the commands are available from the core!
- Update-Help command installs the documentation to the local computer!
 - More useful is the online help!
- <https://technet.microsoft.com/en-us/library/hh847741.aspx>
 - PS 5.0



Thank you!