

PowerShell: Security

Inhaltsverzeichnis

Execution Policy.....	2
Commands.....	2
Anpassungsmöglichkeiten.....	2
Scope-Werte.....	2
Execution Context	3
Commands.....	3
Language Modes	3
Anpassungsmöglichkeiten.....	3
Profile	4
Commands.....	4
Profile-Types.....	4
Anpassungsmöglichkeiten.....	5
Einschränkung per Language-Mode erteilen:	5
Remoting	6
Commands.....	6
WinRM Service - Properties	6
Firewall Rules.....	7
Commands.....	7
Anpassungsmöglichkeiten.....	7
Session Configuration.....	8
Commands.....	8
Anpassungsmöglichkeiten.....	8

Execution Policy

Commands

```
Get-Executionpolicy
```

```
Get-Executionpolicy -list
```

Anpassungsmöglichkeiten

Scope für...

```
Scope ExecutionPolicy
-----
MachinePolicy } GPO
UserPolicy   }
Process      } Set-ExecutionPolicy
CurrentUser  }
LocalMachine }
```

Scope-Werte¹

AllSigned	Nur signierte Scripts werden ausgeführt, das gilt auch für lokal erstellte.
RemoteSigned	Aus dem Internet heruntergeladene Scripts müssen signiert sein
Restricted	Erlaubt individuelle Befehle, aber keine Scripts. Wird als Standardwert gesetzt.
Unrestricted	Alle Scripts werden ausgeführt. Bei nicht signierten Scripts aus dem Internet muss man jede Ausführung am Prompt bestätigen.
Bypass	Keinerlei Einschränkungen, Warnungen oder Prompts.
Undefined	Entfernt eine zugewiesene Richtlinie

Mehr auf: docs.microsoft.com

¹ windowspro.de | docs.microsoft.com

Execution Context

Commands

```
$ExecutionContext.SessionState.LanguageMode
```

Language Modes²

FullLanguage	Erlaubt alle Language-Elemente in der Session. Ist als Standard-Wert gesetzt.
ConstrainedLanguage	Erlaubt alle cmdlet- und PowerShell-Elemente. Es gibt jedoch eine beschränkte Anzahl Typen.
RestrictedLanguage	Es können cmdlet-Befehle etc. erlaubt werden, jedoch keine Script-Blöcke.
NoLanguage	Es ist keine Form von Script erlaubt. Es muss mit der «AddCommand()» oder «AddParameter()» gearbeitet werden.

Anpassungsmöglichkeiten

Angenommen «FullLanguage» ist gesetzt. Um die Benutzung so einzuschränken, dass man jedoch trotzdem noch Scripts schreiben kann, könnte man den Language-Mode auf «ConstrainedLanguage» setzen. Damit ist die Anzahl Typen, welche bei PowerShell verwendet werden können, beschränkt.

² docs.microsoft.com

Profile

Commands

```
Test-Path -Path $PROFILE.AllUsersAllHosts
```

```
Test-Path -Path $PROFILE.AllUsersCurrentHost
```

```
...USW.
```

Profile-Types³

All Users, All Hosts (Console)	Betrifft alle Benutzer auf allen Geräten in der PowerShell-Konsole.
All Users, Current Host (Console)	Betrifft alle Benutzer auf dem aktuellen Gerät in der PowerShell-Konsole.
Current User, All Hosts (Console)	Betrifft den aktuellen Benutzer auf allen Geräten in der PowerShell-Konsole.
Current User, Current Host (Console)	Betrifft den aktuellen Benutzer auf dem aktuellen Gerät in der PowerShell-Konsole.
All Users, Current Host (ISE)	Betrifft alle Benutzer auf allen Geräten in der PowerShell-ISE.
Current User, Current Host (ISE)	Betrifft den aktuellen Benutzer auf dem aktuellen Gerät in der PowerShell-ISE.

Hinweis:

Es setzt bei *All Users/Hosts* voraus, dass die jeweiligen Benutzer & Geräte auf den Pfad des Profils zugreifen können. Ein Profil besteht aus einem .ps1-File.

³ devblogs.microsoft.com

Anpassungsmöglichkeiten

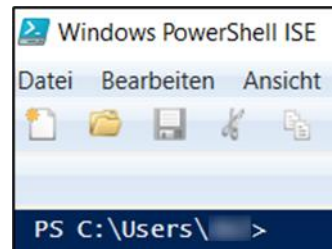
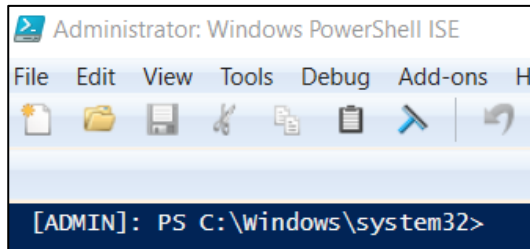
Man könnte ein «Admin-Tag» vor den Pfad setzen, damit man direkt sieht, ob man sich als Admin in PowerShell befindet.



Create_ISE-Profile.ps1 4

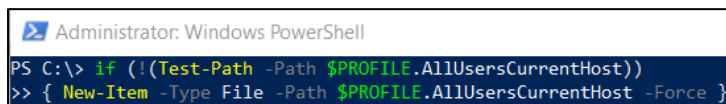


Create_Admin-Tag.ps1 5

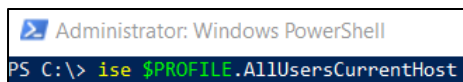


Einschränkung per Language-Mode erteilen:

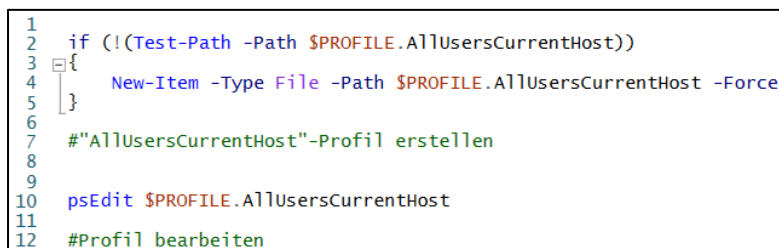
1.



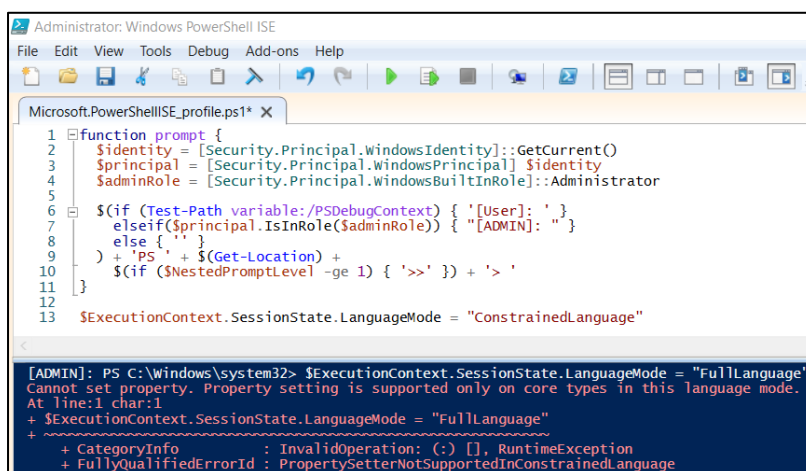
2.



3.



4.



⁴ docs.microsoft.com

⁵ docs.microsoft.com

Remoting

Commands

<code>Get-Service WinRM SELECT *</code>
<code>winrm quickconfig</code>
<code>ls WSMAN:\localhost\Service Select-Object -Property "Name","Value"</code>

WinRM Service - Properties⁶

RootSDDL	Definiert die Zugriffsberechtigungen für Remoting. (Default = <i>O:NSG:BAD:P(A;;;GA;;;BA)(A;;;GR;;;ER)S:P(AU;FA;GA;;;WD)(AU;SA;GWGX;;;WD)</i>)
MaxConcurrentOperations MaxConcurrentOperationsPerUser	Die maximale Anzahl gleichzeitiger Operationen. (Default = 100) Die maximale Anzahl gleichzeitiger Operationen, welche jeder Benutzer per Remote auf demselben System öffnen kann. (Default = 1'500)
EnumerationTimeoutms	Definiert den Leerlauf-Time-Out (in Millisekunden) zwischen Pull-Nachrichten (Default = 60'000)
MaxConnections	Definiert die maximale Anzahl der aktiven Abfragen, welche der Service simulativ produzieren kann. (Default = 300)
MaxPacketRetrievalTimeSeconds	Definiert die maximale Länge der Zeit (in Sek.), welche WinRM benötigt, um ein Paket zu erhalten. (Default = 120)
AllowUnencrypted	Der Client kann unverschlüsselten Traffic abfragen. (Default = <i>False</i>)
DefaultPorts	Definiert den http-/https-Port für den Client. (Default http = 5985 Default https = 5986)
IPv4Filter IPv6Filter	Definiert die benutzbaren Adressen. (Default IPv4 = *) (Default IPv6 = *)
EnableCompatibilityHttpListener EnableCompatibilityHttpsListener	Definiert ob der HTTP-/HTTPS-Listener zusätzlich zum Port 5985/5986 auf den Port 80/443 hören soll. (Default http = <i>False</i>) (Default https = <i>False</i>)
AllowRemoteAccess	Definiert den Remote-Zugriff auf den Client. (Default = <i>False</i>)

⁶ docs.microsoft.com

Firewall Rules

Commands

```
Get-NetFirewallRule -Name "WinRM" | Select-Object -Property "Name","Displayname","Enabled"
```

```
New-NetFirewallRule -DisplayName "example" -Name "example" -Profile "Public" -Enabled "True"
```

Anpassungsmöglichkeiten⁷

Ports deaktivieren...	Nicht nötig, wenn bereits alle deaktiviert sind.
Firewall Rules hinzufügen	(Siehe Beispiel)

```
PS C:\> New-NetFirewallRule -DisplayName "example" -Name "example" -Profile "Public" -Enabled "True"

Name                : example
DisplayName          : example
Description         :
DisplayGroup        :
Group               :
Enabled             : True
Profile             : Public
Platform            : {}
Direction           : Inbound
Action              : Allow
EdgeTraversalPolicy  : Block
LooseSourceMapping  : False
LocalOnlyMapping    : False
Owner               :
PrimaryStatus       : OK
Status              : The rule was parsed successfully from the store. (65536)
EnforcementStatus   : NotApplicable
PolicyStoreSource    : PersistentStore
PolicyStoreSourceType : Local
```

⁷ docs.microsoft.com

Session Configuration

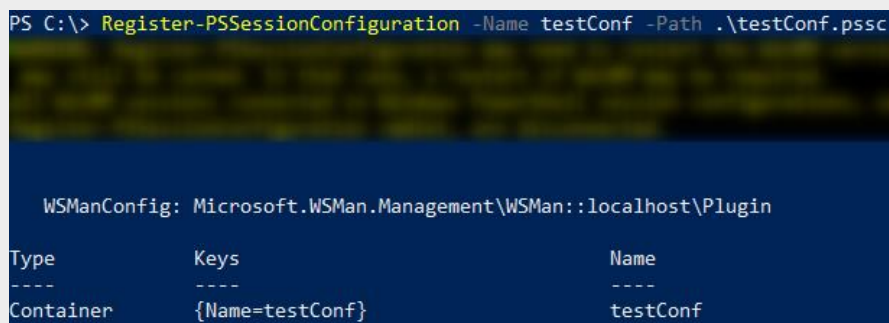
Commands

Get-PSSessionConfiguration

Anpassungsmöglichkeiten⁸

Eine neue Session-Konfiguration erstellen

```
PS C:\> Register-PSSessionConfiguration -Name testConf -Path .\testConf.pssc
```



Type	Keys	Name
Container	{Name=testConf}	testConf

Zugriff auf cmdlet-Ausdrücke einschränken

```
PS C:\> New-PSSessionConfigurationFile -LanguageMode ConstrainedLanguage -Path ".\testConf.pssc" -VisibleCmdlets "Get", "Select"
```

Eine Möglichkeit, solche cmdlet-Ausdrücke einzuschränken, bietet das JEA-Prinzip. JEA steht für «Just Enough Administration», also für limitierte Berechtigungen.⁹

Beispiel:

MitarbeiterX arbeitet im Bereich SW-Paketierung und MitarbeiterY im Bereich Bitlocker. Beide besitzen Administratoren-Rechte, würden jedoch lediglich 65% davon benötigen.

Hier könnte man also per JEA-Prinzip die Admin-Rechte auf die benötigten 65% einschränken.

⁸ docs.microsoft.com | sid-500.com

⁹ security-insider.de