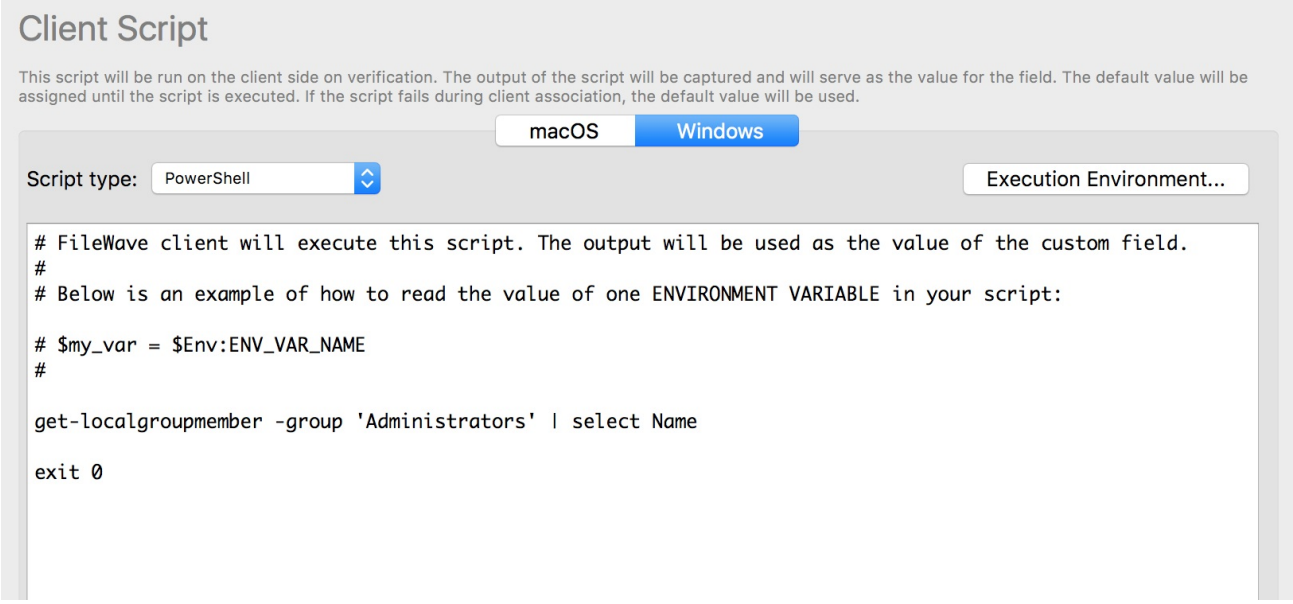
Running Built-in PowerShell Commands with Custom Fields

In Filewave when using custom fields for PowerShell scripts the commands are run in a 32-bit environment. Some built-in PowerShell commands require 64-bit to work. In order to run the commands you need to modify the command to launch the correct PowerShell environment.

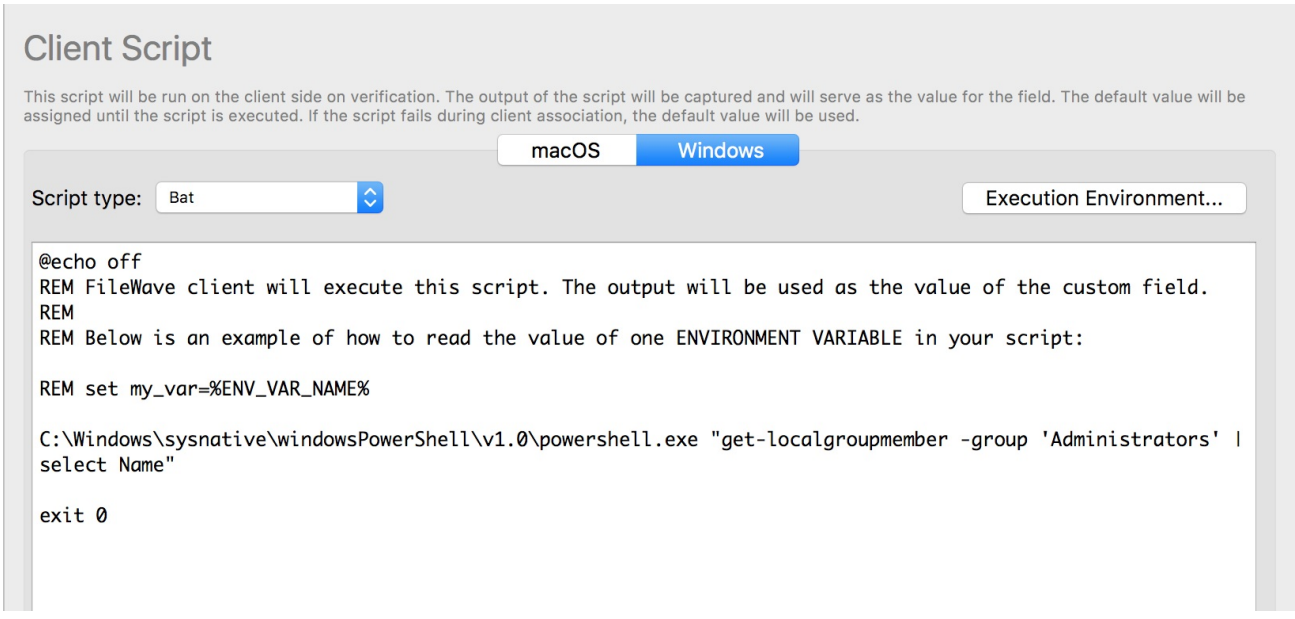
An example of this is with the below PowerShell command to list the local Admin accounts on a given device. The screen shot below shows how you would normally setup the Custom Field in your environment. The issue is that this command needs to be run in a 64 bit environment while FileWave defaults to 32 bit when executing the commands.

```
get-localgroupmember -group 'Administrators' | select Name
```



Since this is the case you will now need to modify the command choosing to run in 64 bit. To do this you would add "C:\Windows\system32\windowsPowerShell\v1.0\powershell.exe" to the command and run it as a Bat script instead of PowerShell in the Custom Field. I posted the code that would work with the Custom Fields below as well as a screen shot of the correct setup.

```
C:\Windows\system32\windowsPowerShell\v1.0\powershell.exe "get-localgroupmember -group 'Administrators' | select Name"
exit 0
```



Another method to execute an entire script in the native environment (32bit on 32bit or 64bit on 64bit) is as follows:

```
################################################################################
#If Powershell is running the 32-bit version on a 64-bit machine, we
#need to force powershell to run in 64-bit mode .
################################################################################
if ($env:PROCESSOR_ARCHITEW6432 -eq "AMD64") {
    #write-warning "Take me to 64-bit....."
    if ($myInvocation.Line) {
        &"$env:WINDIR\system32\windowspowershell\v1.0\powershell.exe" -NonInteractive -NoProfile
$myInvocation.Line
    }else{
        &"$env:WINDIR\system32\windowspowershell\v1.0\powershell.exe" -NonInteractive -NoProfile -file
"$($myInvocation.InvocationName)" $args
    }
exit $lastexitcode
}

# Main script
# Uncomment the next line to prove that we are always in 64bit
#[Environment]::Is64BitProcess

# Your 64bit script here.

################################################################################
#End
################################################################################
```

Related Content

- Scripting Languages supported in FileWave

Revision #1
★Created 2 July 2023 14:12:47 by Josh Levitsky
Updated 2 July 2023 14:33:05 by Josh Levitsky