# Default waarden instellen

$minuten = 1;

$server = "localhost";

$home = "c:\gebruikers\administrator\bureaublad\"

$whiteListFile = $home + '\MonTool\WhiteList.txt';

$signaleringenFile = $home + '\MonTool\Signaleringen.txt';

# Ask for user input for server name and minutes

$server = Read-Host -Prompt "Enter the server name"

$minutes = Read-Host -Prompt "Enter the number of minutes"

# Check if the entered server is valid

$checkServer = Test-Connection -ComputerName $server -Count 1 -Quiet

# Use the user input in the script

Write-Host "The server name is: $server"

Write-Host "The number of minutes is: $minutes"

# Tijdsinterval voorbereiden

$periode = New-timespan -minutes $minuten;

$startTijd = Get-Date;

$stopTijd = (Get-Date) + $periode;

# Check if the whitelist and signaleringen files exist

If (!(Test-Path $whiteListFile) -or !(Test-Path $signaleringenFile)) {

Write-Host "One or both of the required files do not exist. Please check the file paths and try again."

Return

}

# Lege signaleringenfile maken

Set-Content -path $signaleringenFile -value $server, $datum;

# Whitelist inlezen in een array

[string[]]$arrayMetWhiteList = Get-Content -Path $whiteListFile

$periode = New-timespan -minutes $minuten; # bereid de tijdsinterval voor

$startTijd = Get-Date;

$stopTijd = (Get-Date) + $periode;

Write-Host "`nProcesChecker is gestart om", $startTijd, "op server", $server, "voor een periode van", $minuten, "minuten";

Set-Content -path $signaleringenFile -value $server, $datum; # maak lege signaleringenfile

[string[]]$arrayMetWhiteList = Get-Content -Path $whiteListFile # lees de witelist in een array

# \*\*\* BEGIN - zorg dat de onderstaande acties herhaald worden tot het $minuten is verstreken \*\*\*

[string[]]$arrayMetProcessen = invoke-command -Computer $server {Get-Process | Select-Object -Expandproperty name}

Foreach ($ProcessName in $arrayMetProcessen) {

If (-Not ($arrayMetWhiteList -Match $processName)) { # proces in de WhiteList?

$answer = Read-Host -Prompt "Proces", $processName, "is onbekend, toevoegen aan whitelist (y/n)?";

While (($answer -ne "y") -And ($answer -ne "n")) {

$answer = Read-Host -Prompt "Keuze (y/n)";

}

If ($answer -Eq 'y') { # procesnaam toevoegen aan whitelist

$arrayMetWhiteList += $processName;

}

Else { # procesnaam toevoegen aan signaleringen

Add-Content -Path $signaleringenFile -Value $processName

}

}

}

# \*\*\* zorg dat ik iets op het scherm zie bewegen gedurende de 5 seconden tussen de herhalingen \*\*\*

Wait-Event -TimeOut 5 # heartbeat, wachtperiode tussen de herhalingen in

# Zet de starttijd

$start = Get-Date

# Zet de eindtijd op 5 seconden na de starttijd

$end = $start.AddSeconds(5)

# Zolang de huidige tijd voor de eindtijd is

while ((Get-Date) -lt $end) {

# Laat iets op het scherm bewegen

$spinner = @("-","\","|","/")

$i = 0

$counter = 0

$colors = @("Yellow","Green","Red","Blue", "white")

$startTime = Get-Date

while ((New-TimeSpan -Start $startTime -End (Get-Date)).TotalSeconds -lt 5) {

$color = $colors[$counter % 4]

Write-Host -NoNewLine -ForegroundColor $color $spinner[$i % 4]

$i++

Start-Sleep -Milliseconds 150

Clear-Host

$counter++

}

}

# \*\*\* EIND - tot hier herhalen gedurende de $minuten \*\*\*

# nu komen de afsluitende acties, whitelist naar bestand zetten en signaleringen printen

# \*\*\* vraag of de whitelist leeg gemaakt moet worden, accepteer alleen y en n \*\*\*

# \*\*\* als de whitelist leeg moet schrijf een lege whitelist weg, anders de huidige \*\*\*

$arrayMetWhiteList | Out-File $whiteListFile;

$clearWhitelist = Read-Host -Prompt "Wilt u de whitelist leegmaken (y/n)?";

While (($clearWhitelist -ne "y") -And ($clearWhitelist -ne "n")) {

$clearWhitelist = Read-Host -Prompt "Keuze (y/n)";

}

If ($clearWhitelist -Eq 'y') {

# maak de whitelistfile leeg

Set-Content -Path $whiteListFile -Value "";

}

Else {

# schrijf de gewijzigde whitelist naar het bestand

Set-Content -Path $whiteListFile -Value $arrayMetWhiteList;

}

# \*\*\* vraag of de signaleringen op het scherm moeten komen, accepteer y om te printen, iets anders zie je als n \*\*\*

# \*\*\* indien y: druk de signaleringen netjes af op het scherm \*\*\*

#Write-Host "`n`n======================`nSignaleringen van", $startTijd, "tot", $stopTijd;

#write-host " \*\*\* deze lijst moet je dus nog maken \*\*\* "

$showSignaleringen = Read-Host -Prompt "Wilt u de signaleringen op het scherm tonen (y/n)?";

While (($showSignaleringen -ne "y") -And ($showSignaleringen -ne "n")) {

$showSignaleringen = Read-Host -Prompt "Keuze (y/n)?";

}

If ($showSignaleringen -Eq 'y') {

# print de signaleringen naar het scherm

Get-Content -Path $signaleringenFile | Write-Host;

}

# Lees de signaleringen in uit het bestand

$signaleringen = Get-Content $signaleringenFile

# Druk elke signalering af op het scherm

$signaleringen | ForEach-Object {

Write-Host " - $\_"

}

# Maak een variabele met de huidige datum/tijd

$endTime = Get-Date

# Voeg de einddatum/tijd toe aan de afsluitende boodschap

Write-Host "======================`nProcesChecker is beëindigd om $endTime`n"