

Comment:

```
# Scriptnaam: ProcessCheck Sjabloon.txt
# functie : check processen van een server tegen een whitelist
# Auteur : Stichting PraktijkLeren, BvdF
# Versie : 0.5
# datum : 23 december 2021
#
# Argumenten [minuten-numerieke string [servernaam-alfanumerieke string]]
#
# de comments die beginnen en eindigen met *** zijn aanwijzingen voor jouw opdracht
#
$minuten = 1; # Zetten van default waarden
$server = "DC01";
$whiteListFile = $home + '\MonTool\WhiteList.txt';
$signaleringsFile = $home + '\MonTool\Signalerings.txt';

# *** zorg dat als het aantal minuten wordt meegegeven bij de start deze de default waarde
# vervangt ***
# *** zorg dat als een servernaam wordt meegegeven bij de start dat die de default waarde
# overschrijft ***
Do {
    $name = Read-Host "Welke server wordt er gebruikt?"
} Until ($name -eq 'DC01')
#Gr@pp1g

$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 $signaleringsFile -value $server, $datum; # maak lege signaleringsfile
[string[]]$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 $signaleringsFile -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

If ($tijdsduur = replyhost "Hoelang moet het process lopen?") {$tijdsduur} else {1}

$a = 9
```

```

Do
{
    "Starting loop $a"
$a
$a ++
    "Now $a is $a"
}
While ($a -le 5)

# *** 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 ***
$Confirmation = Read-Host -Prompt "Wil je de whitelist leeg maken? [Y/N]"
while ($Confirmation -ne 'y'){
    if ($Confirmation -eq 'n') {{exit}}
    $Confirmation = Read-Host "Wil je de whitelist leeg maken? [Y/N]"
}
},
#GR@pp1g

# *** als de whitelist leeg moet schrijf een lege whitelist weg, anders de huidige ***
$arrayMetWhiteList | Out-File $whiteListFile;

# *** 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 ***
$Confirmation = Read-Host -Prompt "Wil je signaleringen op het scherm weergeven? [Y/N]"
while ($Confirmation -ne 'n'){
    if ($Confirmation -eq 'y') {
        $Confirmation = Write-Host "`n`n=====`nSignaleringen van", $startTijd,
"tot", $stopTijd;
        Write-Host " *** deze lijst moet je dus nog maken *** "
    }
}
},
#GR@pp1g

# *** voeg de einddatum/tijd toe aan de afsluitende boodschap
Write-Host "=====`nProcesChecker is beëindigd om $(Get-Date).", "`n" ;
#GR@pp1g

```