Hengst, Quinten van

Ontwerp Casus C

Versie 1.0

Inhoudsopgave

[Doel 2](#_Toc153448290)

[Plan van aanpak 2](#_Toc153448291)

[Eisen 2](#_Toc153448292)

[Wensen 2](#_Toc153448293)

[Uitvoering 2](#_Toc153448294)

[Eindresultaat 2](#_Toc153448295)

[Stroomdiagram 3](#_Toc153448296)

[Uitleg stroomdiagram 3](#_Toc153448297)

[Stappen 4](#_Toc153448298)

# Doel

De klant waarvoor wij werken heeft de volgende probleemstelling; “bepaalde windows-services vormen een beveiligingsrisico wanneer deze actief zijn. Het handmatig controleren en uitschakelen van deze services neemt veel tijd in beslag en kan fouten bevatten.” Wij als IT-bedrijf hebben als doel dit probleem bij de klant weg te nemen en alle wensen, eisen en behoeftes die de klant heeft, waar te maken. Dit doen we door een script aan te leveren die het zo veilig mogelijk maakt om Windows-machines na te kijken op actieve Windows-services en deze uit te schakelen wanneer nodig.

# Plan van aanpak

Het bedrijf waar ik voor werk heeft een strikt beleid betreft te beveiliging, bepaalde Windows-services mogen namelijk niet actief zijn, aangezien deze een beveiligingsrisico vormen. Doordat het handmatig nakijken tijdrovend en foutgevoelig is, is het belangrijk dat dit geautomatiseerd word.

## Eisen

Het script moet voldoen aan de volgende eisen;

* Alle paden zijn relatief en niet hardcoded
* Controleer of alles van tevoren werkt, zoals:

A. Kun je services beheren/bekijken op het systeem?

B. Bestaat het blacklist-bestand?

C. Het script stopt met uitvoeren als deze tests allemaal mislukken en een passende melding maken.

D. Het script moet met één commando direct worden aangeroepen, bijvoorbeeld Check-Services. De naamgeving van het PowerShell-script moet voldoen aan de richtlijnen van Microsoft.

## Wensen

* PowerShell script moet digitaal ondertekend zijn

## Uitvoering

Hier onder is te lezen welke activiteiten er moeten worden uitgevoerd voor het scrip.

* Documenten opstellen
* Flowchart aanmaken
* Script maken
* Script testen
* Eindresultaat op leveren

## Eindresultaat

Als eindresultaat word er het volgende opgeleverd;

* Een overzichtelijke flowchart
* Bijbehorende documenten
* Een getest en werkend script

# Stroomdiagram

A diagram of a workflow

Description automatically generated

## Uitleg stroomdiagram

Uit dit stroomdiagram is te lezen welke stappen het script ondergaat. Zo zijn er meerdere uitkomsten van het script en is het duidelijk welke meldingen er worden gegenereerd.

## Stappen

Het script word gestart

Stap 1: Powershell controleert of de juiste versie gebruikt word. Dit is belangrijk om over de juiste functies te beschikken.

Stap 1a/b: Op het moment dat er een onjuiste versie van powershell word gebruikt, zou deze een melding terug geven met de tekst: “Onjuiste versie gedetecteerd” Nadat deze melding gegeven is gaat het script zichzelf beëindigen.

Stap 2: Deze stap is te zien als de powershell versie check (stap 1) juist is en geeft vervolgens de melding: “Juiste versie gedetecteerd.”

Stap 3: In deze stap worden alle actieve services opgevraagd en opgeslagen in een variable.

Stap 4: In deze stap word er nagekeken of de zwartelijst.txt bestaat. (Het controleren van een bestaand bestand is cruciaal om ervoor te zorgen dat er op een later moment geen error terug gegeven word.)

Stap 4a/b: Op het moment dat het bestand niet gevonden is, geeft het script de melding: “Zwartelijst.txt is niet gevonden” Hierna word het script beëindigd.

Stap 5: Nadat de zwartelijst.txt gevonden is gaat het script door met het controleren van de actieve services, de actieve services worden uit de variable (stap 3) opgevraagd en vergeleken met de services op de zwartelijst.

Stap 5a: Op het moment dat er een service actief is en op de zwartelijst staat, word deze uigeschakeld.

Stap 6: Deze stap geeft een melding terug met de tekst: “Service check voltooid”. Deze stap word direct zichtbaar op het moment dat alle services al uitstonden die inactief moeten zijn.

Stap 7: Het script sluit af.