# **Full System Scan Implementation Guide.**

**Overview**

This document outlines the steps to automate security tasks, including **scheduling full-system scans, enabling real-time protection, configuring Windows Task Scheduler to run scans every 12 hours, and emailing scan logs to the security team**. These steps will ensure continuous protection and monitoring of security threats.

**1. PowerShell Script Functionality**

This PowerShell script performs the following security tasks:

* **Enables Windows Defender Real-Time Protection** to ensure active monitoring.
* **Schedules a full-system scan** at a predefined interval (every 12 hours).
* **Logs scan results** and captures detected threats.
* **Emails scan logs** using Outlook’s SMTP server to the security team.

⚠**Security Considerations**

⚠ **Avoid storing plaintext passwords** in scripts. Instead, use PowerShell’s **SecureString** for safer password handling.

# Convert the plain text password "your-password" to a secure string

$SecurePassword = ConvertTo-SecureString "your-password" -AsPlainText -Force

# Create a new PSCredential object using the provided username and secure password

$Creds = New-Object System.Management.Automation.PSCredential ($Username, $SecurePassword)

**PowerShell Script**

## ===========================================================================

## NAME:        FullSystemScan.ps1

## CREATED:     06-MAR-2025

## BY:          DAVID RADOICIC

## VERSION:     1.0

## DESCRIPTION: Checks the Windows system logs for detected threats and emails reports.

##

##

## NOTE:

## Enables Windows Defender Real-Time Protection to ensure active monitoring.

## Schedules a full-system scan at a predefined interval (every 12 hours).

## Logs scan results and captures detected threats.

## Emails scan logs using Outlook’s SMTP server to the security team.

## ===========================================================================

# Enable Windows Defender Real-Time Protection

Set-MpPreference -DisableRealtimeMonitoring $false

# Define Email Parameters for Outlook SMTP

$SMTPServer = "smtp.office365.com"

$SMTPPort = "587"

$From = "your-email@outlook.com"  # Replace with your Outlook email

$To = "YOUR EMAIL HERE"

$Subject = "Daily Windows Defender Scan Report"

$Username = "your-email@outlook.com"  # Your Outlook username

$Password = "your-secure-password"  # Your Outlook email password (consider using SecureString)

# It is recomended to us use PowerShell’s SecureString for safer password handling, rather than plain text

# Convert the plain text password "your-password" to a secure string

# $SecurePassword = ConvertTo-SecureString "your-password" -AsPlainText -Force

# Create a new PSCredential object using the provided username and secure password

# $Creds = New-Object System.Management.Automation.PSCredential ($Username, $SecurePassword)

# Define Log File Path

$LogFile = "C:\DefenderScanReport.log"

# Start Full Scan & Log Output

Start-MpScan -ScanType FullScan | Out-File -FilePath $LogFile -Append

Start-Sleep -Seconds 1800  # Wait for scan to complete (adjust timing if needed)

# Collect Windows Defender Threat Log

$ThreatLog = Get-MpThreatDetection | Format-Table -AutoSize | Out-String

Add-Content -Path $LogFile -Value "`nThreat Detection Summary:`n$ThreatLog"

# Send Email with Scan Report via Outlook SMTP

$Body = Get-Content -Path $LogFile | Out-String

$SMTPClient = New-Object Net.Mail.SmtpClient($SMTPServer, $SMTPPort)

$SMTPClient.EnableSsl = $true

$SMTPClient.Credentials = New-Object System.Net.NetworkCredential($Username, $Password)

$MailMessage = New-Object System.Net.Mail.MailMessage($From, $To, $Subject, $Body)

$SMTPClient.Send($MailMessage)

# Confirm Email Sent

Write-Output "Defender Scan Log sent to $To"

# Schedule Daily Scan Task at 2 AM

$action = New-ScheduledTaskAction -Execute "PowerShell.exe" -Argument "-Command Start-MpScan -ScanType FullScan"

$trigger = New-ScheduledTaskTrigger -Daily -At 2am

$principal = New-ScheduledTaskPrincipal -UserId "NT AUTHORITY\SYSTEM" -LogonType ServiceAccount

$settings = New-ScheduledTaskSettingsSet -AllowStartIfOnBatteries -DontStopIfGoingOnBatteries -StartWhenAvailable

Register-ScheduledTask -TaskName "DailyFullSystemScan" -Action $action -Trigger $trigger -Principal $principal -Settings $settings

Write-Output "Scheduled a full system scan daily at 2 AM."

⚠ **If you prefer to schedule the job through Task Scheduler comment out the Daily Scan Task code. Then follow the below instructions.** ⚠

**2. Scheduling the Script to Run Every 12 Hours Using Task Scheduler**

To automate the execution of this script, configure Windows Task Scheduler to run it every 12 hours.

**Step 1: Open Task Scheduler**

1. Press Win + R, type taskschd.msc, and press **Enter**.
2. Click **Create Basic Task** in the right-hand panel.

**Step 2: Set Task Name and Description**

1. Name the task **"Windows Defender Scheduled Scan"**.
2. Add a description: *"Runs a full-system scan and emails security logs every 12 hours."*.
3. Click **Next**.

**Step 3: Set Trigger (Schedule Task Every 12 Hours)**

1. Select **Daily** and click **Next**.
2. Set the **start time** (e.g., 6:00 AM) and select **Recur every 1 days**.
3. Click **Next**.
4. Select **Repeat Task Every 12 Hours**.

**Step 4: Set Action (Run PowerShell Script)**

1. Select **Start a Program** and click **Next**.
2. In the **Program/Script** field, type:
3. powershell.exe
4. In the **Add Arguments** field, enter:
5. -File "C:\Path\To\Your\Script.ps1"
6. Click **Next**.

**Step 5: Set Additional Security Options**

1. Select **Run whether user is logged on or not**.
2. Check **Run with highest privileges**.
3. Click **Finish**.

**Step 6: Test Task Execution**

1. Right-click the created task and select **Run**.
2. Verify that the scan starts and completes successfully.
3. Check your **Outlook Sent Items** for the email log.

**3. Security Enhancements & Best Practices**

To further improve security:

* Use **SecureString** for storing email passwords securely.
* Set up **email alerts for failed scans**.
* Regularly review **Defender threat detection logs**.
* Ensure **Windows Defender definitions** are always up to date.

**Conclusion**

This automation setup ensures continuous protection by: ✔ Running a full-system scan every 12 hours. ✔ Logging and capturing potential threats. ✔ Emailing scan logs to the security team for review. ✔ Enhancing overall security monitoring.

By implementing this script and scheduled task, On Time Decorations will maintain **proactive cybersecurity defenses** with minimal manual intervention. 🚀