SCRIPT TO MONITOR LOGS

AIM:

Create a script that monitors server logs for errors and alerts you.

INTRODUCTION:

Server logs are records of events generated by a server, capturing requests, errors, security events, and system performance. They help administrators monitor, troubleshoot, and optimize server operations. Monitoring server logs is essential for maintaining system health, ensuring security, troubleshooting issues, and optimizing performance. It involves collecting, analyzing, and responding to log data in real-time to detect anomalies, failures, or security threats.

OBJECTIVE:

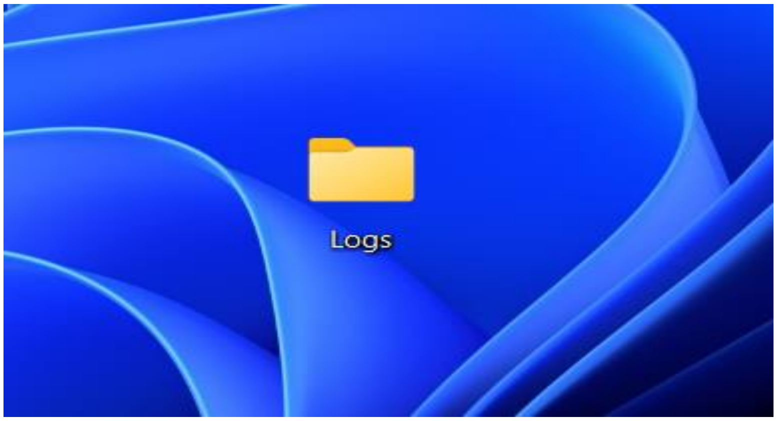
The objective of this project is to:

- Automate the process of monitoring log files for critical events.
- Learn how to create and execute PowerShell scripts on a
- Windows system.
- Enhance troubleshooting efficiency by providing immediate
- alerts for critical events.

STEP BY STEP OVERVIEW:

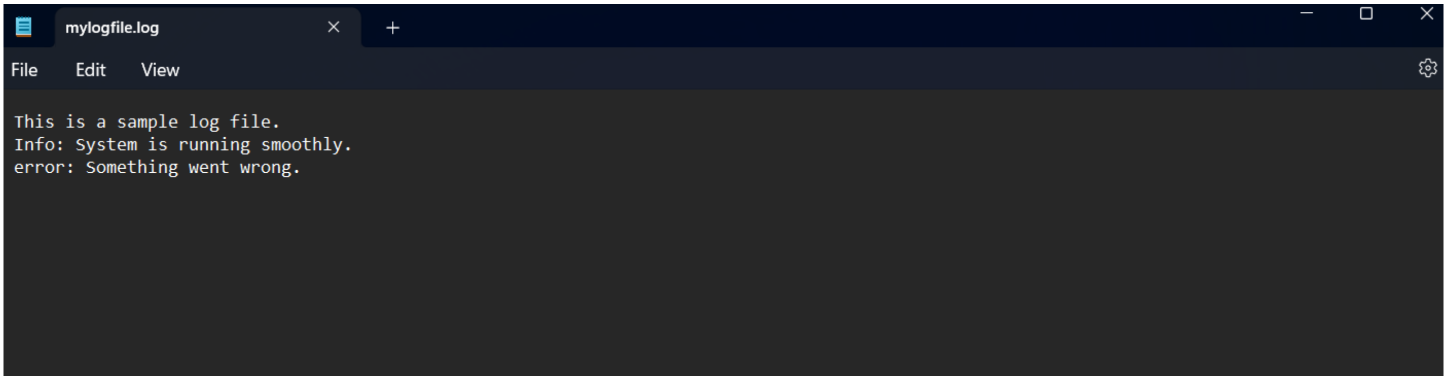
Step 1: CREATE A FOLDER

- Create a folder named ‘Logs’ for your logs and script.



Step 2: EDIT LOG FILE

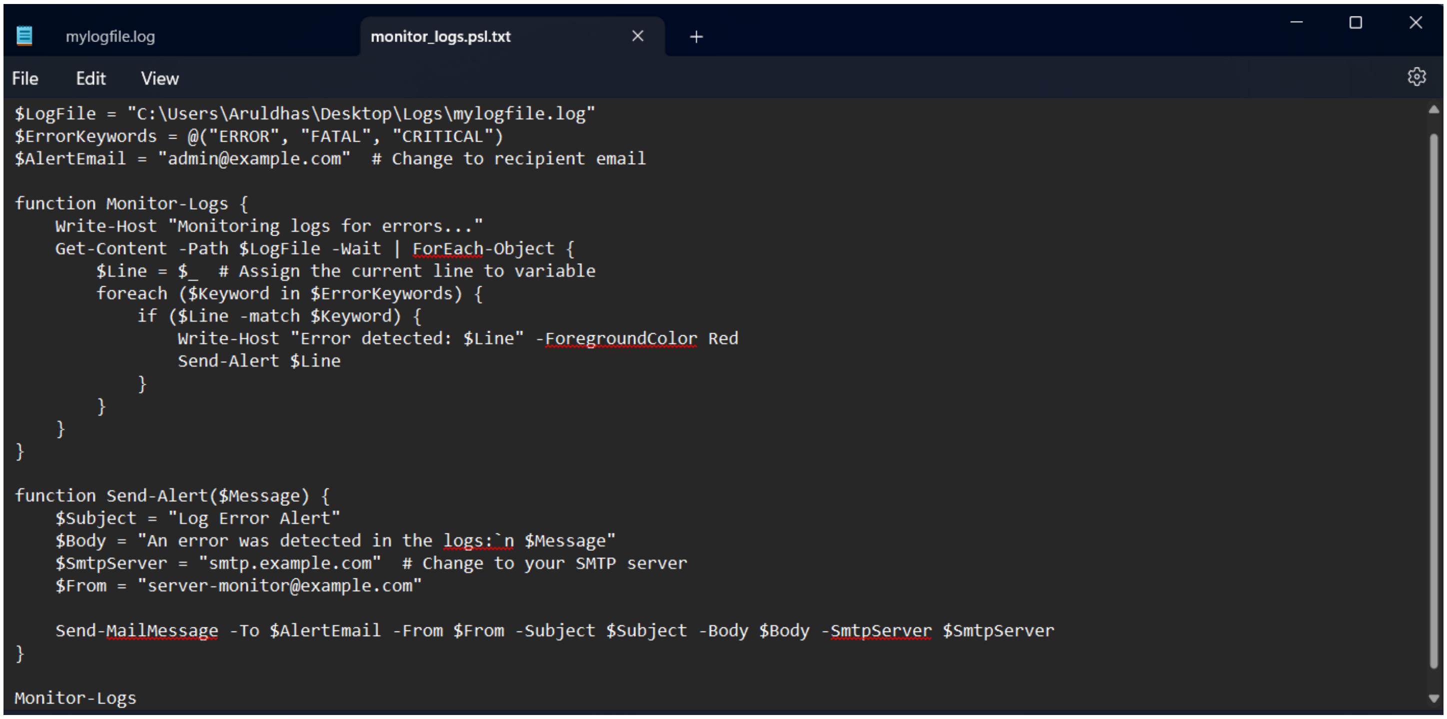
- Open your Notepad and add the following sample test to it and save the file as ‘mylogfile.log’ in the logs folder.



```
This is a sample log file.
Info: System is running smoothly.
error: Something went wrong.
```

Step 3: WRITE SCRIPT IN POWERSHELL.

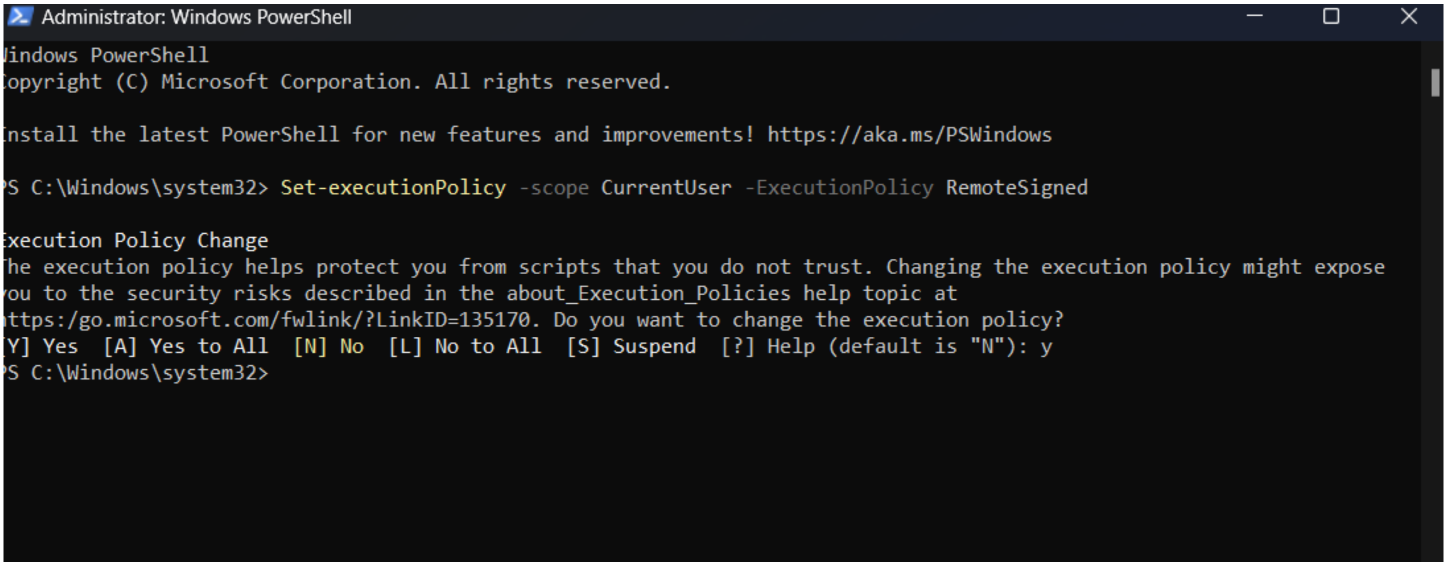
- Open your Notepad and write the following PowerShell script into it and set the ‘Log file path address’ to the mylogfile.log.
- Save the file as ‘monitor_logs.psl’ inside the same logs folder.



```
$LogFile = "C:\Users\Aruldhas\Desktop\Logs\mylogfile.log"
$ErrorKeywords = @("ERROR", "FATAL", "CRITICAL")
$AlertEmail = "admin@example.com"  # Change to recipient email

function Monitor-Logs {
    Write-Host "Monitoring logs for errors..."
    Get-Content -Path $LogFile -Wait | ForEach-Object {
        $Line = $_  # Assign the current line to variable
        foreach ($Keyword in $ErrorKeywords) {
            if ($Line -match $Keyword) {
                Write-Host "Error detected: $Line" -ForegroundColor Red
                Send-Alert $Line
            }
        }
    }
}

function Send-Alert($Message) {
    $Subject = "Log Error Alert"
    $Body = "An error was detected in the logs:`n $Message"
    $SmtpServer = "smtp.example.com"  # Change to your SMTP server
    $From = "server-monitor@example.com"

    Send-MailMessage -To $AlertEmail -From $From -Subject $Subject -Body $Body -SmtpServer $SmtpServer
}

Monitor-Logs
```

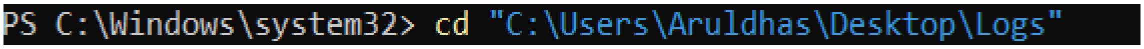
Step 4: SCRIPT EXECUTION

- Open PowerShell and click run as ‘administrator’.
- Write the following command to allow script execution.



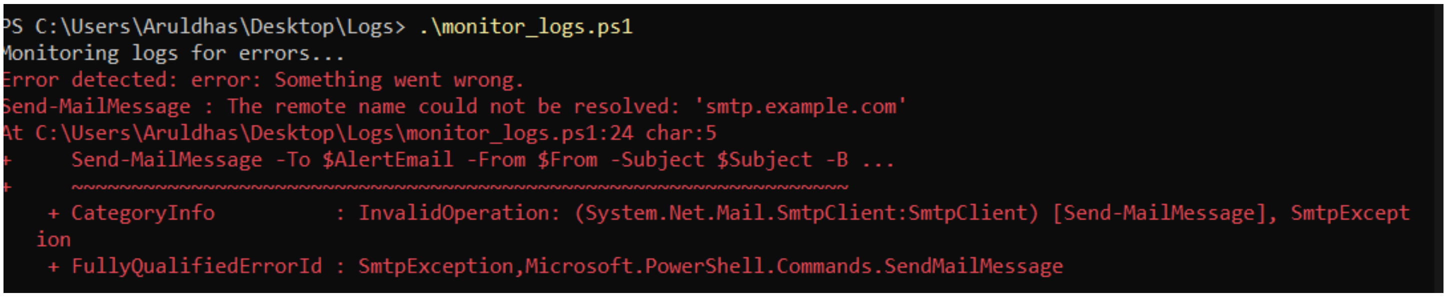
```
Administrator: Windows PowerShell
Jindows PowerShell
opyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Windows\system32> Set-executionPolicy -scope CurrentUser -ExecutionPolicy RemoteSigned

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
ou to the security risks described in the about_Execution_Policies help topic at
ttps:/go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): y
PS C:\Windows\system32>
```

- Navigate to the logs folder.



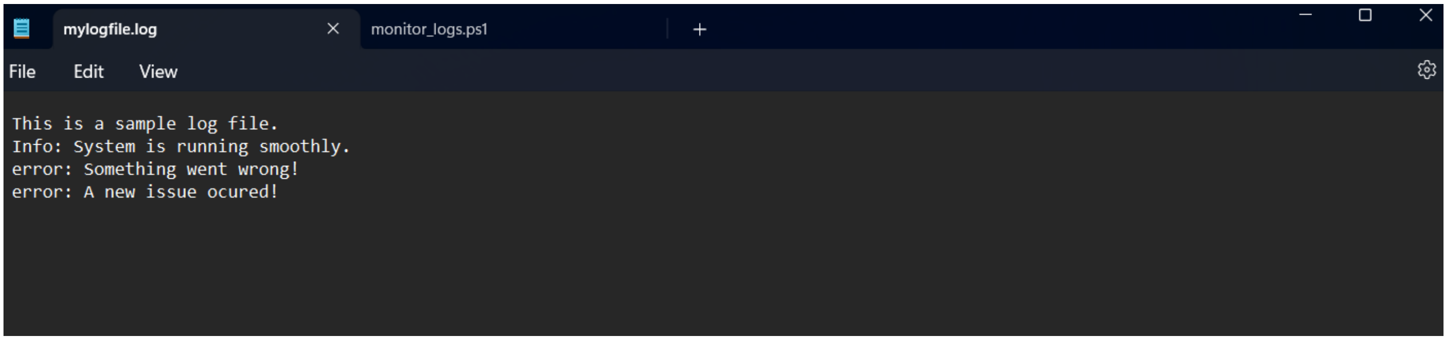
```
PS C:\Windows\system32> cd "C:\Users\Aruldhas\Desktop\Logs"
```

- Now, run the script by using the command ‘.\monitor_logs.ps1’



```
PS C:\Users\Aruldhas\Desktop\Logs> .\monitor_logs.ps1
Monitoring logs for errors...
Error detected: error: Something went wrong.
Send-MailMessage : The remote name could not be resolved: 'smtp.example.com'
At C:\Users\Aruldhas\Desktop\Logs\monitor_logs.ps1:24 char:5
+     Send-MailMessage -To $AlertEmail -From $From -Subject $Subject -B ...
+     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (System.Net.Mail.SmtpClient:SmtpClient) [Send-MailMessage], SmtpExcept
   ion
    + FullyQualifiedErrorId : SmtpException,Microsoft.PowerShell.Commands.SendMailMessage
```

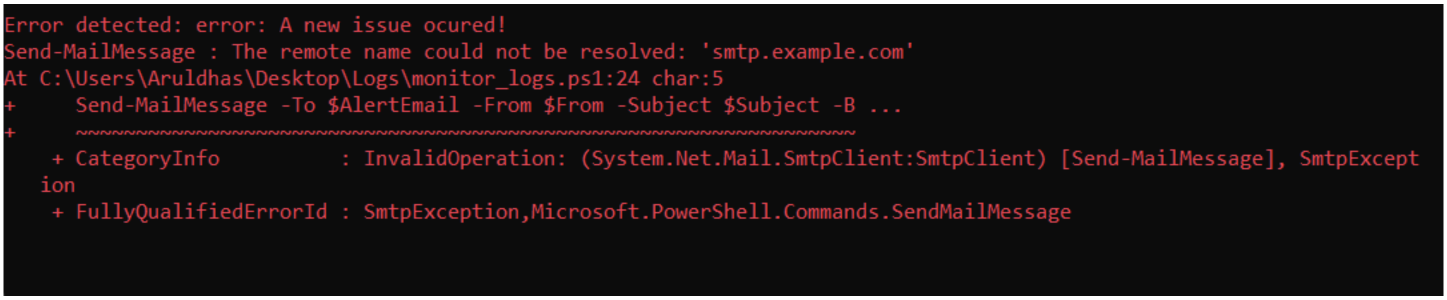
- You will see an error message occurred. Now, open your Notepad and add a new line with ‘error’ message.



```
This is a sample log file.
Info: System is running smoothly.
error: Something went wrong!
error: A new issue ocured!
```

Step 5: ALERT MESSAGE

- Now you will see an alert message with the ‘error’ keyword.
- Eg: error: A new issue occurred!



```
Error detected: error: A new issue ocured!
Send-MailMessage : The remote name could not be resolved: 'smtp.example.com'
At C:\Users\Aruldhas\Desktop\Logs\monitor_logs.ps1:24 char:5
+     Send-MailMessage -To $AlertEmail -From $From -Subject $Subject -B ...
+     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (System.Net.Mail.SmtpClient:SmtpClient) [Send-MailMessage], SmtpExcept
   ion
    + FullyQualifiedErrorId : SmtpException,Microsoft.PowerShell.Commands.SendMailMessage
```

CONCLUSION:

Thus, by completing this POC, we have understood:

- To successfully create and execute the script that monitors the server logs.
- Monitors and detect the error which then send alert on predefined keywords.
- The importance of log monitoring in system maintenance.