CHAPS (Hardening Assessment PowerShell Script) Assignment Report

Prepared by: ABINESHKUMAR MUTHUSAMY

Date: 23/02/2024

Client: AK

Executive Summary:

The CHAPS assessment was conducted on the systems belonging to AK to evaluate the security posture and identify potential vulnerabilities. This report provides an overview of the findings and recommendations for improving the security of the systems.

Assessment Overview:

The assessment using CHAPS have performed various tests and checks to harden the powershell and suggested us some fixes to be performed.

The assessment covered the following areas:

- Windows Security Settings and configurations Patch Management
- User Account Settings and Permissions
- Group Policy Settings
- Firewall Configurations
- Common Security Vulnerabilities

STEPS FOR RUNNING THE CHAPS:

1. Open Github and download the code for running CHAPS(Configuration Hardening Assessment and Powershell Script).
2. Now open another system or OS connected in the network and run http server. Open Linux terminal and run **python3 –m http.server 8181**. This will start the web server listening on all of the system's IP addresses.
3. Now open command prompt on the target system with administrator rights and run the code **powershell.exe -exec bypass** to being a PowerShell prompt.
4. Then after beginning the PS prompt **Set-ExecutionPolicy Bypass -scope Process** to allow scripts to execute.
5. From this prompt, run the following command to execute the chaps.ps1 script using **./chaps.ps1.**
6. Each script's outputs will be written to the user's **Temp** directory as defined by the $env:temp variable. Copy these files off of the system being reviewed and delete them.
7. Then go to run the CHAPS Powersploit Security checks. The PowerSploit project (dev

branch) can be used to gather additional information about the system. The chaps-powersploit.ps1 script has been developed to gather this information. Of course, most anti-malware programs will prevent, protect, and alert on the use of PowerSploit. Therefore, the anti-malware should be disabled or the chaps-powersploit.ps1 script should not be used.

The assessment performed several checks and tests. Some of the important checks and tests are summarized:

1. Checking for missing Windows patches with Critical or Important MsrcSeverity values: It is important to keep the operating system and all installed software up-to-date with the latest security patches to reduce the risk of vulnerabilities being exploited.
2. Checking BitLocker Encryption: BitLocker encryption helps protect data at rest by encrypting the entire drive, making it difficult for attackers to access sensitive information even if they gain physical access to the device.
3. Checking if users can install software as NT AUTHORITY\SYSTEM: Allowing users to install software as the NT AUTHORITY\SYSTEM account can be a significant security risk, as it grants them administrative privileges and could allow them to install malicious software.
4. Testing if PowerShell Commandline Auditing is Enabled: Enabling PowerShell command line auditing can help detect and investigate potential security incidents by logging all PowerShell commands executed on the system.
5. Testing if PowerShell Moduling is Enabled: Enabling PowerShell module logging can help detect and investigate potential security incidents by logging all PowerShell modules loaded on the system.
6. Testing if PowerShell EnableScriptBlockLogging is Enabled: Enabling PowerShell script block logging can help detect and investigate potential security incidents by logging all PowerShell scripts executed on the system.
7. Testing if PowerShell EnableTranscripting is Enabled: Enabling PowerShell transcripting can help detect and investigate potential security incidents by logging all PowerShell sessions and commands executed during those sessions.
8. Testing if PowerShell EnableInvocationHeader is Enabled: Enabling PowerShell invocation header can help detect and investigate potential security incidents by logging the source of PowerShell scripts and commands.
9. Testing if PowerShell ProtectedEventLogging is Enabled: Enabling PowerShell protected event logging can help detect and investigate potential security incidents by logging all PowerShell events that are considered security-relevant.
10. Checking event logs settings defaults are too small: Ensuring that event logs are configured with sufficient maximum sizes can help ensure that important security-related events are not overwritten before they can be reviewed and investigated.
11. Testing if PowerShell is configured to use Constrained Language: Configuring PowerShell to use constrained language can help prevent the execution of potentially malicious PowerShell scripts.
12. Testing if system is configured to limit the number of stored credentials: Limiting the number of stored credentials can help reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
13. Testing if system is configured to prevent RDP service: Preventing remote access via

Terminal Services can help reduce the risk of attackers gaining unauthorized access to sensitive systems and data.

14. Testing if WinFW Service is running: The Windows Firewall service is an important security control that helps protect the system from unauthorized network access.
15. Testing if Windows Network Firewall rules allow remote connections: Ensuring that Windows Network Firewall rules are configured to allow only necessary remote connections can help reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
16. Testing Local Administrator Accounts: Ensuring that there are no unnecessary local administrator accounts can help reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
17. Testing if AppLocker is configured: AppLocker is a powerful security control that can help prevent the execution of unauthorized software on the system.
18. Testing if system is configured to enable WINS Resolution: Disabling WINS resolution can help reduce the risk of attackers using NetBIOS to gain unauthorized access to sensitive systems and data.
19. Testing if Net Session Enumeration configuration using the TechNet script NetSessEnumPerm.ps1: Ensuring that Net Session Enumeration is configured securely can help reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
20. Testing for WPAD entry in C:\Windows\System32\Drivers\etc\hosts: Ensuring that there is no WPAD entry in the hosts file can help prevent attackers from using WPAD to perform man-in-the-middle attacks.
21. Testing for WPADOverride registry key: Ensuring that the WPADOverride registry key is not configured can help prevent attackers from using WPAD to perform man-in-the-middle attacks.
22. Testing WinHttpAutoProxySvc configuration: Ensuring that the WinHttpAutoProxySvc service is not running can help reduce the risk of attackers using WPAD to perform man-in-the-middle attacks.

Findings and Recommendations:

1. IPv6 Network Settings: The system has an IPv6 network interface assigned, which could be a potential security risk. It is recommended to disable IPv6 if it is not required.
2. Missing Critical or Important Update KB: The system is missing a critical or important update KB5034441. It is recommended to install the latest updates to ensure the system is secure.
3. BitLocker Encryption: BitLocker is not detected on the operating system volume or encryption is not complete. It is recommended to enable BitLocker encryption to protect data at rest.
4. PowerShell Commandline Audting: ProcessCreationIncludeCmdLine_Enabled is not set. It is recommended to enable PowerShell command line auditing to detect and investigate potential security incidents.
5. PowerShell Moduling: EnableModuleLogging is not set. It is recommended to enable PowerShell module logging to detect and investigate potential security incidents.

6. PowerShell EnableScriptBlockLogging: EnableScriptBlockLogging is not set. It is recommended to enable PowerShell script block logging to detect and investigate potential security incidents.
7. PowerShell EnableScriptBlockInvocationLogging: EnableScriptBlockInvocationLogging is not set. It is recommended to enable PowerShell script block invocation logging to detect and investigate potential security incidents.
8. PowerShell EnableTranscripting: EnableTranscripting is not set. It is recommended to enable PowerShell transcripting to detect and investigate potential security incidents.
9. PowerShell EnableInvocationHeader: EnableInvocationHeader is not set. It is recommended to enable PowerShell invocation header to detect and investigate potential security incidents.
10. PowerShell ProtectedEventLogging: EnableProtectedEventLogging is not set. It is recommended to enable PowerShell protected event logging to detect and investigate potential security incidents.
11. Event logs settings defaults are too small: The maximum sizes for various event logs are smaller than recommended. It is recommended to increase the maximum sizes for event logs to ensure important security-related events are not overwritten before they can be reviewed and investigated.
12. PowerShell Version: The system is running PowerShell version 5.1.19041.3803, which is below the recommended version 5. It is recommended to upgrade to the latest version of PowerShell.
13. PowerShell is configured to use Constrained Language: Execution Language Mode is not set to ConstrainedLanguage. It is recommended to configure PowerShell to use constrained language to prevent the execution of potentially malicious PowerShell scripts.
14. System is configured to limit the number of stored credentials: CachedLogonsCount is not set to 0 or 1. It is recommended to limit the number of stored credentials to reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
15. Computer Browser service is running: It is recommended to disable the Computer Browser service to reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
16. NetBios is enabled: It is recommended to disable NetBios to reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
17. Security back-port patch KB2871997 is not installed: It is recommended to install the latest security back-port patch to ensure the system is secure.
18. SMBv1 is enabled: It is recommended to disable SMBv1 to reduce the risk of attackers gaining unauthorized access to sensitive systems and data.
19. Lanman Authentication for LM Compatability Level registry key is not configured: It is recommended to configure the LM Compatability Level registry key to require NTLMv2 and 128-bit encryption.
20. Domain and Local Anonymous Enumeration settings: RestrictAnonymous registry key is

not configured: It is recommended to configure the RestrictAnonymous registry key to restrict anonymous enumeration.

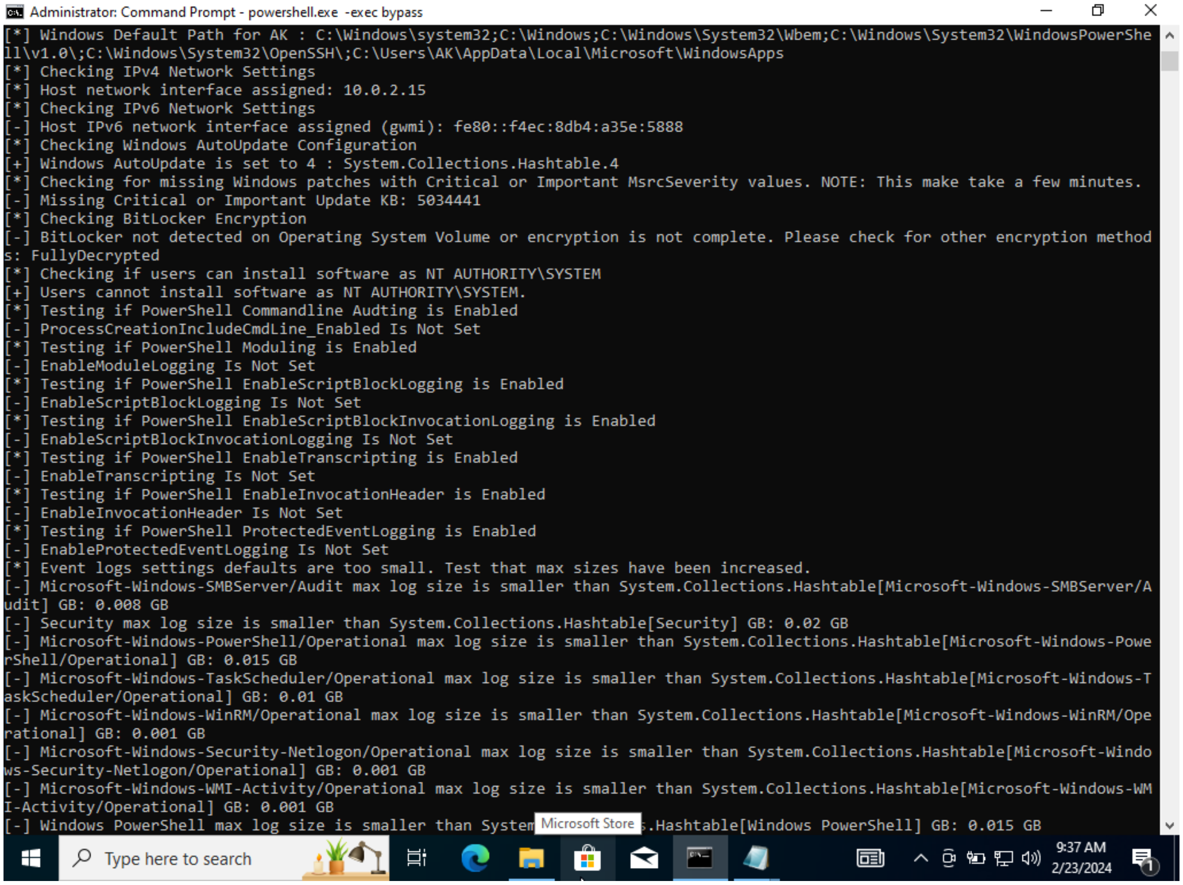
21. NTLM Session Server Security settings and NTLM Session Client Security settings are not configured to require NTLMv2 and 128-bit encryption: It is recommended to configure NTLM Session Server Security settings and NTLM Session Client Security settings to require NTLMv2 and 128-bit encryption to reduce the risk of attackers gaining unauthorized access to sensitive systems and data.

Conclusion:

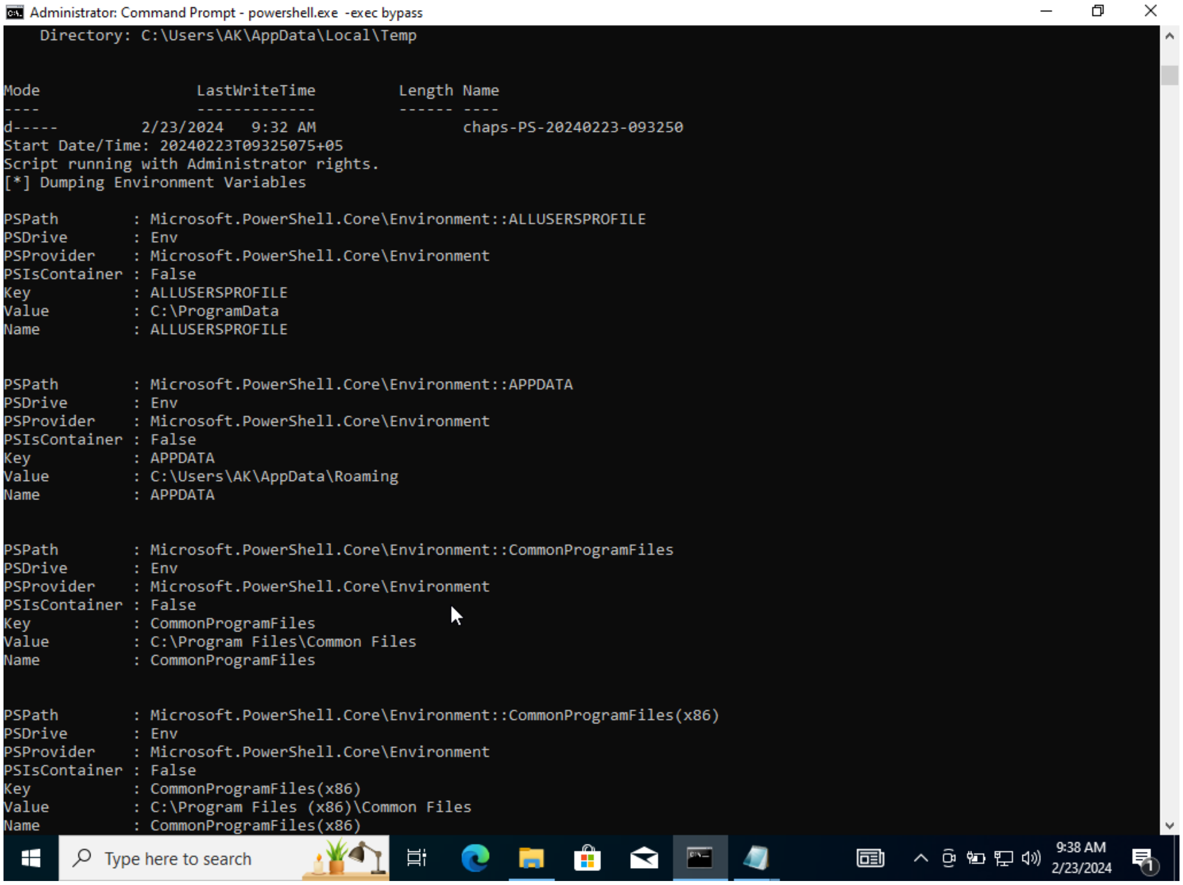
The CHAPS assessment identified several areas where improvements can be made to enhancethe security posture of AK's system. By implementing the recommendations outlined in this report, AK can reduce the risk of security breaches and protect sensitive data from unauthorized access.

This concludes the CHAPS Hardening Assessment Report for AK.

THE SAMPLE SCREENSOT TAKEN WHEN THE PROCESS RAN IS ATTACHED HERE FOR YOUR REFERENCE:



```
[*] Windows Default Path for AK : C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShe
ll\v1.0\;C:\Windows\System32\OpenSSH\;C:\Users\AK\AppData\Local\Microsoft\WindowsApps
[*] Checking IPv4 Network Settings
[*] Host network interface assigned: 10.0.2.15
[*] Checking IPv6 Network Settings
[-] Host IPv6 network interface assigned (gwmi): fe80::f4ec:8db4:a35e:5888
[*] Checking Windows AutoUpdate Configuration
[+] Windows AutoUpdate is set to 4 : System.Collections.Hashtable.4
[*] Checking for missing Windows patches with Critical or Important MsrcSeverity values. NOTE: This make take a few minutes.
[-] Missing Critical or Important Update KB: 5034441
[*] Checking BitLocker Encryption
[-] BitLocker not detected on Operating System Volume or encryption is not complete. Please check for other encryption method
s: FullyDecrypted
[*] Checking if users can install software as NT AUTHORITY\SYSTEM
[+] Users cannot install software as NT AUTHORITY\SYSTEM.
[*] Testing if PowerShell Commandline Audting is Enabled
[-] ProcessCreationIncludeCmdLine_Enabled Is Not Set
[*] Testing if PowerShell Moduling is Enabled
[-] EnableModuleLogging Is Not Set
[*] Testing if PowerShell EnableScriptBlockLogging is Enabled
[-] EnableScriptBlockLogging Is Not Set
[*] Testing if PowerShell EnableScriptBlockInvocationLogging is Enabled
[-] EnableScriptBlockInvocationLogging Is Not Set
[*] Testing if PowerShell EnableTranscripting is Enabled
[-] EnableTranscripting Is Not Set
[*] Testing if PowerShell EnableInvocationHeader is Enabled
[-] EnableInvocationHeader Is Not Set
[*] Testing if PowerShell ProtectedEventLogging is Enabled
[-] EnableProtectedEventLogging Is Not Set
[*] Event logs settings defaults are too small. Test that max sizes have been increased.
[-] Microsoft-Windows-SMBServer/Audit max log size is smaller than System.Collections.Hashtable[Microsoft-Windows-SMBServer/A
udit] GB: 0.008 GB
[-] Security max log size is smaller than System.Collections.Hashtable[Security] GB: 0.02 GB
[-] Microsoft-Windows-PowerShell/Operational max log size is smaller than System.Collections.Hashtable[Microsoft-Windows-Powe
rShell/Operational] GB: 0.015 GB
[-] Microsoft-Windows-TaskScheduler/Operational max log size is smaller than System.Collections.Hashtable[Microsoft-Windows-T
askScheduler/Operational] GB: 0.01 GB
[-] Microsoft-Windows-WinRM/Operational max log size is smaller than System.Collections.Hashtable[Microsoft-Windows-WinRM/Ope
rational] GB: 0.001 GB
[-] Microsoft-Windows-Security-Netlogon/Operational max log size is smaller than System.Collections.Hashtable[Microsoft-Windo
ws-Security-Netlogon/Operational] GB: 0.001 GB
[-] Microsoft-Windows-WMI-Activity/Operational max log size is smaller than System.Collections.Hashtable[Microsoft-Windows-WM
I-Activity/Operational] GB: 0.001 GB
[-] Windows PowerShell max log size is smaller than System               s.Hashtable[Windows PowerShell] GB: 0.015 GB
```



```
    Directory: C:\Users\AK\AppData\Local\Temp


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         2/23/2024   9:32 AM                chaps-PS-20240223-093250
Start Date/Time: 20240223T09325075+05
Script running with Administrator rights.
[*] Dumping Environment Variables

PSPath        : Microsoft.PowerShell.Core\Environment::ALLUSERSPROFILE
PSDrive       : Env
PSProvider    : Microsoft.PowerShell.Core\Environment
PSIsContainer : False
Key           : ALLUSERSPROFILE
Value         : C:\ProgramData
Name          : ALLUSERSPROFILE


PSPath        : Microsoft.PowerShell.Core\Environment::APPDATA
PSDrive       : Env
PSProvider    : Microsoft.PowerShell.Core\Environment
PSIsContainer : False
Key           : APPDATA
Value         : C:\Users\AK\AppData\Roaming
Name          : APPDATA


PSPath        : Microsoft.PowerShell.Core\Environment::CommonProgramFiles
PSDrive       : Env
PSProvider    : Microsoft.PowerShell.Core\Environment
PSIsContainer : False
Key           : CommonProgramFiles
Value         : C:\Program Files\Common Files
Name          : CommonProgramFiles


PSPath        : Microsoft.PowerShell.Core\Environment::CommonProgramFiles(x86)
PSDrive       : Env
PSProvider    : Microsoft.PowerShell.Core\Environment
PSIsContainer : False
Key           : CommonProgramFiles(x86)
Value         : C:\Program Files (x86)\Common Files
Name          : CommonProgramFiles(x86)
```