	DP Security Team Network Defense	**SOP #**	SPP-1509
		Revision #	1
		Implementation Date (MM/DD/YYYY)	02/25/2020
Page #	1 of xx	**Last Reviewed/Update Date**	
SOP Owner	Donald R. Phillips	**Approval**	

SPP-1509: Rackspace Win Server 2019 PCI Hardening

Document Type: Standard Operating Procedure

Table of Contents

SPP-1509: Rackspace Win Server 2019 PCI Hardening 1

Document Type: Standard Operating Procedure 1

1. Purpose – 2

2. Scope – 2

3. Prerequisites – 2

4. Responsibilities – 3

5. Procedure 4

5.1 Introduction 4

5.2 BigFix Procedure 5

5.3 Manual Procedure 7

5.4 Script Explanation (Still in Development) 9

5.5 Upcoming Features (Still in Development) 13

6. References 14

1. Purpose –

Explaining the Windows Server 2019 Rackspace PCI Hardening Script (an updated version of the legacy Lockdown.bat file ran for the legacy Datapipe customers).

2. Scope –

System Admins/Engineers who work with customers to harden their servers according to PCI requisite guidance.

3. Prerequisites –

1. Operating system must be: Windows Server 2019.
2. PowerShell must be: Installed (this is the default from Microsoft, however should be confirmed).
3. (If running through BES task) The BigFix client: must run as admin/system in order to elicit all of the changes.
4. (If running manually) The user running the script: Must be able to run as admin in order to elicit all of the changes.

4. Responsibilities –

1. Must be aware of whether the customer is requesting this for a domain joined machine or not.
    a. Also must be aware of whether this is being run on a Domain Controller or not.
2. (After script is run) Must ensure to check the Windows "Event Viewer" > Applications and Services Logs" > "Rackspace Hardening" log.
    a. This is where all tracking of the script running is logged to. This will report both successful and failure attempts. Failures may be caused due to services not present or various other items, regardless of the reasoning, should be reviewed.
3. (After script is run) Confirm that the hardening Registry Key is present so that the successful application is reported to BigFix for tracking.
    a. 32-Bit Applications: HKEY_LOCAL_MACHINE\Software\Rackspace\Hardening
    b. 64-Bit Applications: HKEY_LOCAL_MACHINE\Software\WOW6432Node\Rackspace\Hardening
4. (After script is run) Confirm that the following folder is present and has backed up information present:
    a. C:\Windows\Datapipe
5. (For Domain Controllers) The script does import two policies for hardening purposes however it does NOT overwrite the Default Domain Policy or the Default Domain Controllers Policy. In order to enact these, the **customer** must decide whether to implement the policies.
    a. See the CIS_Server_2019_GPO.xlsx Excel spreadsheet which outlines which policies were or were not implemented in the policies, the reasonings, and the settings.
    b. They may also choose to copy the GPO and redact some of the settings. This choice **MUST** be made by the customer, as they are according to PCI settings however could potentially impact services/applications.

5. Procedure

5.1 Introduction

This document outlines and explains the WS2019_CIS_SetPolicy_0.x_%.ps1 PowerShell script. This script was created as an update to the legacy "Datapipe_Windows_OS_Hardening"/"Lockdown.bat" script.

The reasons for this update are as follows:

1. The "Lockdown.bat" script is an older script form which is presently outpaced and outdated due to PowerShell becoming the main scripting tool of Windows.
2. The "Datapipe_Windows_OS_Hardening"/"Lockdown.bat" script was not documented at all, and pulled on legacy tools which are either no longer relevant, no longer recognized, or simply have been removed/no longer function in the script any further.
3. The last update of the Lockdown.bat was for Windows Server 2012 R2. This is a much older script which required modifications to address new challenges and issues with new PCI Requirements and System requirements to sufficiently meet our necessary testing & auditing needs.

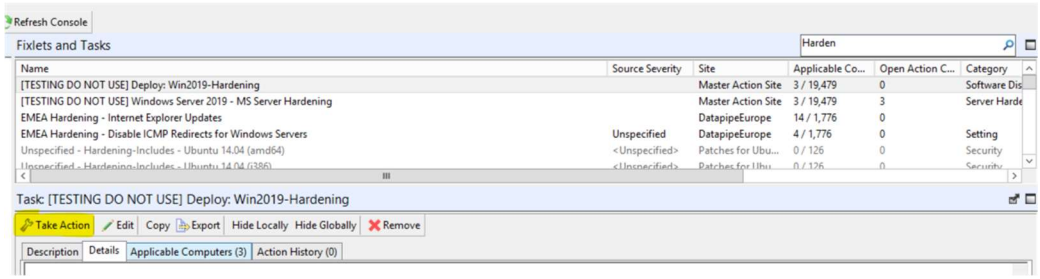
The script itself does have comments throughout (including links for references to certain functions for understanding).

This script does meet all of the same functionality as the "Lockdown.bat" script did, however in an updated manner for ease of documentation, understanding, and functionality.

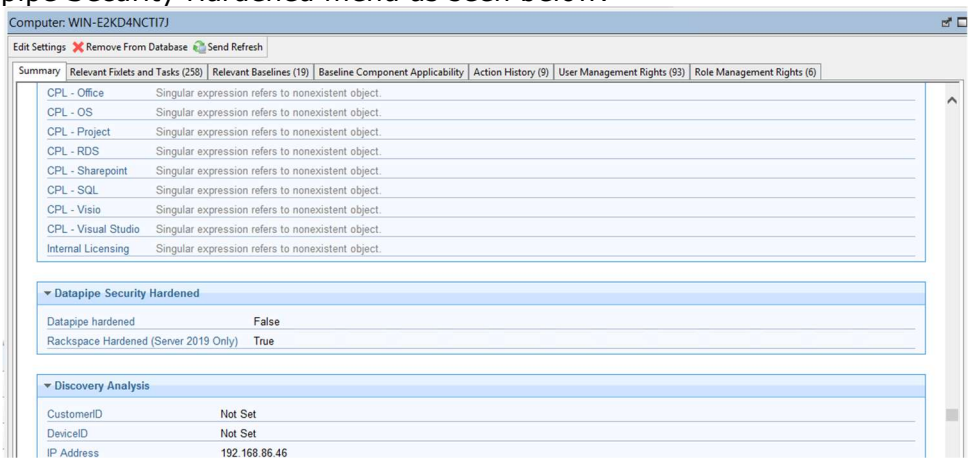
*****Important Note:**

- As of this issue of the Standard Operating Procedure and script, it is still in it's development phase. Although the script has been tested on a virtual machine for both manual and BigFix distributions/implementations, there may be some problems which occur. As those problems are found, please send an email to Donald.Phillips@Rackspace.com in order for the issues to be noted and corrected in follow up versions.
- As of this issue of the Standard Operating Procedure and script, due to the development stage still being active, there are functions and tools within the script which do not function, or have been commented out. While our goal is to make these functions work, there was a limited time to script deployment, and we wanted to address the core functionality first.

5.2 BigFix Procedure

1. If you are trying to automate deployment of this script, first log in to one of the Datapipe BigFix consoles using your Datapipe credentials after logging into one of the SANTA environments.
    a. BESConsole01.datapipe.net
    b. BESConsole02.datapipe.net
2. Once you have successfully RDP'd to one of the above servers, launch the "IBM BigFix Console", and log in with either your Datapipe or Intensive credentials.
3. Once the console has fully loaded (please allow 10-20 minutes for this to occur, select "Fixlets and Tasks" in the top of the "All Content" menu.
4. Once this has loaded, in the top right Search box, type in the correct form of the script which you are attempting to run **This will be updated once further functions within the script are completed**:
    a. Member Servers/Non-Domain Joined Servers/Non-Domain Controllers/Domain Controllers which do **NOT** want to include our PCI GPOs:
    "Windows Server 2019 – MS Server Hardening"
    b. Domain Controllers:
    "Windows Server 2019 – DC Server Hardening"
    c. Member Server, Non-Domain Joined Servers, or Domain Controllers having **ALL** the policies rolled back:
    "Windows Server 2019 – Full Rollback"
    d. Member Server, Non-Domain Joined Servers, or Domain Controllers having **ONLY** the File ACLs, Registry, Windows Services rolled back:
    "Windows Server 2019 – FS Only Rollback"
5. Once you have selected the function you wish to take, summarily click "Take Action" on the task.
    a. 

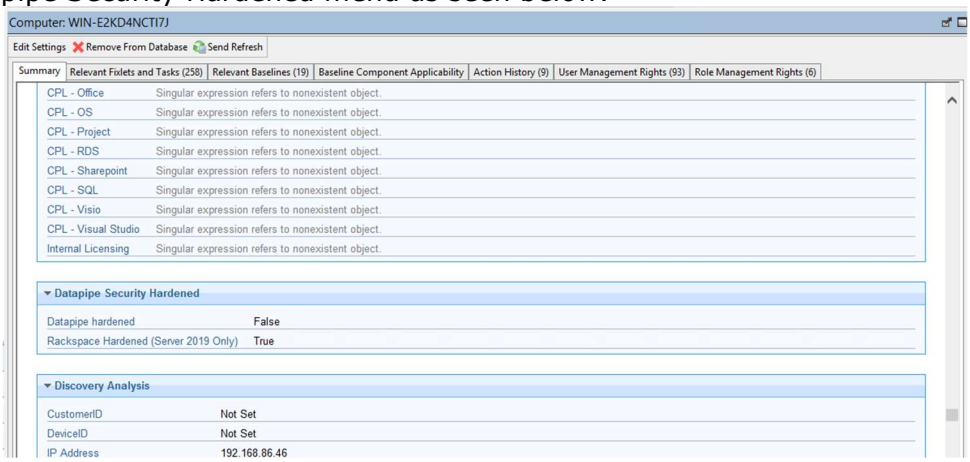
    b. When selecting the action, do **not** select a reboot after, as the reboot could potentially interrupt the script running which can take a fair amount of time.
    c. The machine however **WILL** require a reboot after the script has completed in order to apply some of the settings for local policy, registry, etc.

6. Once the script has completed, you should be able to view the logged actions and successes/failures within check the Windows "Event Viewer" > Applications and Services Logs" > "Rackspace Hardening" log.
7. If there are any issues with running the script, or if services/functions which should have run successfully report an error, please log them in DPOne under the following ticket:
    a. TASK0269771
    b. Direct Ticket Link: TASK0269771
8. Once the script has run, within 24 hours (analysis updates once per day) the hardening should reflect within the BigFix > Computers > %ComputerName% > Datapipe Security Hardened menu as seen below:
    a. 

    b. This analysis is based off of the existence of a Registry key:
        i. 32-Bit Applications: HKEY_LOCAL_MACHINE\Software\Rackspace\Hardening
        ii. 64-Bit Applications: HKEY_LOCAL_MACHINE\Software\WOW6432Node\Rackspace\Hardening
9. (After script is run) Confirm that the following folder is present and has backed up information present:
    a. C:\Windows\Datapipe

5.3 Manual Procedure

1. If you have downloaded the script from BigFix or are planning to download it, the links are as follows:
   **Be aware, the zip file contains extensive files for Domain Controller policies and other tools, the script MUST be run from the same folder that is extracted from the Zip file, otherwise connections will not be made successfully.
   a. BigFix -BigFix Zip File
   b. Share Drive - RAX Global Sharepoint - Shared Link
      i. For this, make sure to select the most recent version. The updates should be listed with a MajorRelease.MinorRelease number "PCI-Hardening_0.6.zip" is the most recent release as of writing this.
2. Once you have downloaded the zip file, copy the Zip over to the computer that you are going to be running it on. Once you extract the folder, ensure that all files and the script reside in the same location.
3. The folder that is extracted will contain a number of different files and folders as well as (at the time of this writing) 3 scripts. Please select the correct script for the function you are trying to complete.
   a. Member Servers/Non-Domain Joined Servers/Non-Domain Controllers/Domain Controllers which do **NOT** want to include our PCI GPOs:
      "WS2019_CIS_SetPolicy_0.6_MS-Hardening.ps1"
   b. Member Server, Non-Domain Joined Servers, or Domain Controllers having **ALL** the policies rolled back:
      "WS2019_CIS_SetPolicy_0.6_RB-Full.ps1"
   c. Member Server, Non-Domain Joined Servers, or Domain Controllers having **ONLY** the File ACLs, Registry, Windows Services rolled back:
      "WS2019_CIS_SetPolicy_0.6_FSOnly.ps1"
   d. Domain Controllers which want to include our PCI GPOs:
      i. At present, there is not a released script for this. Until we create the Default Domain Controllers Group Policy and add further functionality to the script.
4. Each of the scripts performs a number of functions which can be reviewed in:
   a. Section 5.4.
5. Launch PowerShell and run Get-ExecutionPolicy.
   a. If the response from the computer is anything other than "Unrestricted", you cannot simply click to run the script.
   b. You will need to run the following from PowerShell:
   %FolderPathToScript%\%ScriptSelected%.ps1 -executionpolicy Unrestricted -WindowStyle Hidden -NonInteractive

6. Once the script has completed, you should be able to view the logged actions and successes/failures within check the Windows "Event Viewer" > Applications and Services Logs" > "Rackspace Hardening" log.
7. If there are any issues with running the script, or if services/functions which should have run successfully report an error, please log them in DPOne under the following ticket:
    a. TASK0269771
    b. Direct Ticket Link: TASK0269771
8. Once the script has run, within 24 hours (analysis updates once per day) the hardening should reflect within the BigFix > Computers > %ComputerName% > Datapipe Security Hardened menu as seen below:
    a. 

    b. This analysis is based off of the existence of a Registry key:
        i. 32-Bit Applications: HKEY_LOCAL_MACHINE\Software\Rackspace\Hardening
        ii. 64-Bit Applications: HKEY_LOCAL_MACHINE\Software\WOW6432Node\Rackspace\Hardening
9. (After script is run) Confirm that the following folder is present and has backed up information present:
    a. C:\Windows\Datapipe

5.4 Script Explanation (Still in Development)

- Note – If you are curious as to which settings are enacted in the Default Domain Policy or Default Domain Controllers Policy (group policies) that can be found at the following: CIS_Server_2019_GPO.xlsx
- I highly suggest reading through the script as it is commented throughout, and referencing the "References" section either in this document or in the script itself. They contain the same links.
- The below explanations are for the "Function %%" included in the script.

1. Starting from the top of the script, the intro, version of the script, various information for the scripts uses.
2. The references section, as stated above, the references are for various functions within the script which are explained better from the links for understanding their usage and what they are doing.
3. Run-PCIHard –
    a. This is the first function in the script, and at this point is largely commented out due to having some trouble with collocating regular Parameters as well as Dynamic Parameters.
    b. The reason this was commented out was to stop troubleshooting the issues for quicker release of the script. At some point, this will take either command line arguments, and for usage to automate the script from BigFix.
    c. Essentially this function identifies which arguments are passed to the script in order to run the appropriate functions through the rest of the script.
4. Make-Log –
    a. As this sounds like, it is creating a new event log on the computer that it is running on.
    b. Then creating the sources needed to log to the new event log correctly.
5. Log-Info/Warn & Rollback-Info/Warn –
    a. This is creating various functions to both write out to the console and to simultaneously log to the event log with minimal command input and for cleaner scripting.
6. Test-OS –
    a. Exactly as it sounds like, this tests and confirms that the Operating System is Windows Server 2019 before continuing on in the script. Because this script was sculpted around the Server 2019 PCI guidelines, accidentally running it on a non-2019 machine could cause issues.
7. Test-DC –
    a. Exactly as it sounds, this tests the computer to see if the Domain Controller role(s) are installed to determine whether it is a DC or not.

While this will not prevent the Member Server script from running on a Domain Controller (if the customer does not want to import our curated PCI Group Policies), it will prevent the Domain Controller script from running on Member Servers which would cause a number of errors.

8. Menu1/Menu2 –
    a. These are vbscript out-menus which will be functional and usable in a later version of the script. These are essentially just pretty/nice interfaces with buttons that people could use to run the script once all of the functionality issues have been remediated.
    b. Clicking on the buttons would essentially pass in specific arguments to the parameters. Once the parameters are set for the options the user wants to take, the script would run the same as doing so automated through BigFix or using one of the preconfigured scripts.
9. Create-GPSource –
    a. This function creates a backup directory for all of the settings that are exported/backed up during the running of the script.
10. LocalPol-Backup -
    a. This function has 3 componenents:
        i. Export the local group policy settings from: C:\Windows\System32\GroupPolicy
        ii. Export the local Security Policy
        iii. Export the local Audit Policy
    b. All of these are backed up at C:\Windows\Datapipe
11. Backup-Reg –
    a. This backs up a number of keys (listed in the script) which are modified during the running of the script for more hardened settings.
    b. They are backed up to the same Windows\Datapipe folder in multiple .Reg files and then create a merged file with all of the settings compiled for rollback later if needed.
12. Reg-Harden –
    a. Since we just backed up the registry keys, this function pulls a .Reg master key file (located in the script folder at the "Tools" directory.
13. Import-LocalPols -
    a. Since local policies have been backed up, we now import PCI compliant local policies which are more restrictive from the "LocalPolicy" folder within the unpacked Zip file.
14. Disable-Service –
    a. Exactly as this sounds, the function pulls a list of services from the folder which was unpacked (the list is "Services.txt") which can potentially make your server more susceptible.
    b. The script provides read-out in the PowerShell window if you are running it manually, otherwise it will log the attempts to stop & disable services within the event log. It will try 3 times before failing.

15. Start-SmartCard –
    a. Although this could be included in the Services list, due to the importance of this with regards to PCI, we created a separate function for it. This will attempt to set the Smart Card Policy Service to Automatic start, and start the Smart Card Policy Service multiple times. It may fail as it may not be present on the system, or the settings may not allow this. If it fails, this is ok.
16. Set-LocalPerms –
    a. This function sets more restrictive permissions on the Event Logs than default. Making sure that the permissions are set to only allow members of the Administrator group the permissions to read and not modify.
    b. The reason for this is to lower the attack vector of potential bad actors, if one were to gain access to the system and escalate to Administrator privileges, they could potentially remove/delete event log entries or the logs themselves, harming our ability to successfully audit an intrusion.
17. DomainPol-Backup –
    a. This is only used for Domain Controller implementations (not active yet)
    b. Runs a backup of all domain policies that exist within a domain. They can be rolled back or otherwise restored using the "Restore-GPO" command within the GroupPolicy module.
    c. The function does **NOT** delete/remove any policies, just creates a backup in case of any unforeseen activities.
18. ApplyRS-GPO –
    a. Again, only used for Domain Controller implementations (not active yet).
    b. Imports two policies:
        i. RS Default Domain Policy
        ii. RS Default Domain Controllers Policy
    c. The two policies do NOT replace the existing Default Domain Policy or Default Domain Controllers Policy, as they are significantly restrictive and implementation should be left to the customer to decide whether to implement them or not.
    d. If the customer decides to implement them, they can use the policy to overwrite their existing policy or to otherwise link that policy instead of the Default Domain Policy(ies)/in tandem with their Default Domain Policy(ies).

19. Hardening-Complete –
    a. This essentially just writes to the event log and PowerShell screen to let the person know that the script has completed.
    b. This does prompt the user (if running manually) if they would like to reboot the computer now. The user running the script is not REQUIRED to reboot the computer immediately, however certain hardening steps will not be present until reboot is completed.

>>>This is the end of the main Hardening section, below this is the Rollback functions explanation.<<<

20. RBPolicy-Menu1 –
    a. Reference Step 8. Menu1/Menu2
    b. This is the same style of menu, however only utilized if the user is running manually and selects to perform a rollback function rather than a hardening function.
21. Rollback-SecPol –
    a. This section pulls the previously backed up Local Group Policy, Local Security Policy, and Local Audit Policy.
    b. If there are no backups, then this function cannot be completed. Check the Rackspace Hardening Event Log if you do not find any folders/files under C:\Windows\Datapipe\GPOBak for rollback.
22. Rollback-GPO –
    a. If there were Domain GPO's that were backed up previously, this function provides a restoration of those Group Policies to their default location and settings from when they were backed up.
23. Rollback-Reg –
    a. If there were Registry keys which were backed up in initial hardening, this will replace them in the Registry, removing the previous settings.
24. Rollback-Services –
    a. This will utilize the same list from Step 14. Disable Services to attempt to re-enable any services which were disabled previously.
25. RB-Complete –
    a. This essentially just writes to the event log and PowerShell screen to let the person know that the script has completed.
    b. This does prompt the user (if running manually) if they would like to reboot the computer now. The user running the script is not REQUIRED to reboot the computer immediately, however certain rollback steps (Group policy, audit policy, etc.) steps will not be present until reboot is completed.
26. Run-%%% -
    a. This section is simply a conglomeration function which runs the specific tasks which pertain to each Hardening or Rollback procedure.

5.5 Upcoming Features (Still in Development)

- Getting the Default Domain Controller Group Policy created, documented, tested, and released.
    - Subsequently this will also allow us to release a 4th version of the script in preparation for the below.
- Colocating all 4 possible actions to a single script. (instead of separate scripts for different actions for Member Servers Vs. Domain Controllers and Hardening Vs. Rollback.
    - This will also change the BigFix task name to a singular task when it is functional and (much like the legacy Datapipe Hardening task, will prompt you for the action you are trying to take).
- Interactive interface including menus which you can select when running the script manually to select the actions you want to take, and providing readout within the PowerShell window as it goes along while simultaneously logging to the Windows Event Log.
- Updating the script to include/creating a new script for, or updating previous versionings of the Legacy Datapipe Hardening script:
    - Windows Server 2016
    - Windows Server 2012 R2

6. References

1. For your awareness, these same references can also be found in the script itself. They are listed in one of the top Comment sections for reference if needed when running manually.
    a. Note - All folders referenced are either deployed in the same zip/folder, or created within the script:

- **$PSScriptRoot** - https://stackoverflow.com/questions/5466329/whats-the-best-way-to-determine-the-location-of-the-current-powershell-script/5466355
- **Function LogMessage** - https://key2consulting.com/how-to-log-scripts-in-powershell/
- **Menu1/Menu2/RBMenu** - http://serverfixes.com/powershell-radiobuttons
- **Import-LocalPols** - https://community.spiceworks.com/how_to/108867-transfer-all-group-policy-settings-from-one-system-to-another
- **Import-GPO** - https://docs.microsoft.com/en-us/powershell/module/grouppolicy/import-gpo?view=win10-ps
- **Restore-GPO** - https://docs.microsoft.com/en-us/powershell/module/grouppolicy/restore-gpo?view=win10-ps
- **Backup-GPO** - https://docs.microsoft.com/en-us/powershell/module/grouppolicy/backup-gpo?view=win10-ps
- **Function RegHarden** - Reference output from Menu2, selected function will produce variable, referenced for RegHarden
- **Function LocalPerms** - https://stackoverflow.com/questions/21104496/replacing-acl-with-inheritance-using-powershell
- **Function Backup Reg** - https://stackoverflow.com/questions/28076128/powershell-export-multiple-keys-to-one-reg-file
- **Dynamic Parameters** - https://martin77s.wordpress.com/2014/06/09/dynamic-validateset-in-a-dynamic-parameter/
- **Configuring regular and Dynamic Parameters** - https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_functions_advanced_parameters?view=powershell-7
- **Group Policy Explanation** (Excel document with all settings/policies enacted or not enacted listed) - https://raxglobal.sharepoint.com/:x:/s/GTSNetworkSecurityFleetManagement/Eao5rztgCF9Pm_wkxwbCW6kBjNm9AAvX5YKJBaImBW50QA?e=qB6JfQ