**Table of Contents**

[Prerequisites 2](#_Toc28575)

Setup File Script.........................................................................................................................2

Application Validation Script [2](#_Toc8904)

WCF Validation Script……………………………………………………………………….…………………………….....…5

WebServer Validation Script………………………………………………………………….………………………………8

VSCron Jobs Validation Script …………………………………………………………………………………….........................…........………..17

RS Server Validation Script……………………………………………………………………………….............…….17

RD Server Validation Script…………………………………………………………………………………………………18

KMS Server Validation Script………………………………………………………………………………………………20

Prerequisites**:**

* AWS account access required
* Powershell ISE access required

1. **Setup File Script:**

****

* Below is the setup file which will contains all the variable. These variables defines all the application values which needs to be verified in the validation scripts.

**-->**

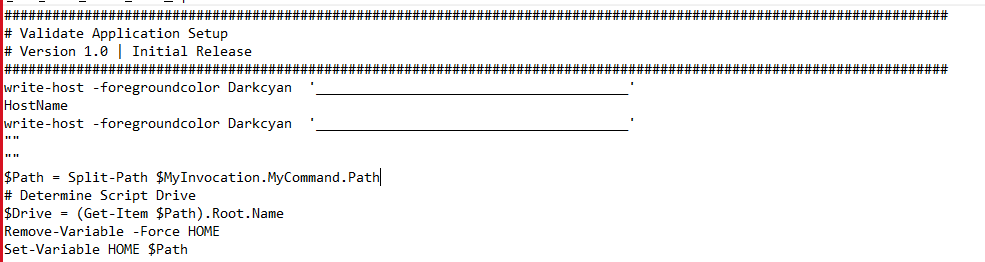


1. **Application Validation Script:**

****

* Below code is used to to keep your Script Dynamic.
* $MyInvocation will store the information of script, How it executed or How script got started/invoked.
* $HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

**-->**

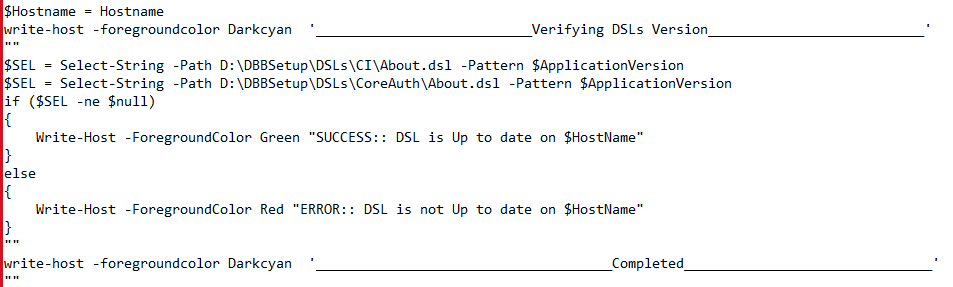


* Below code is used to call variables which stored in setup file

***-->***

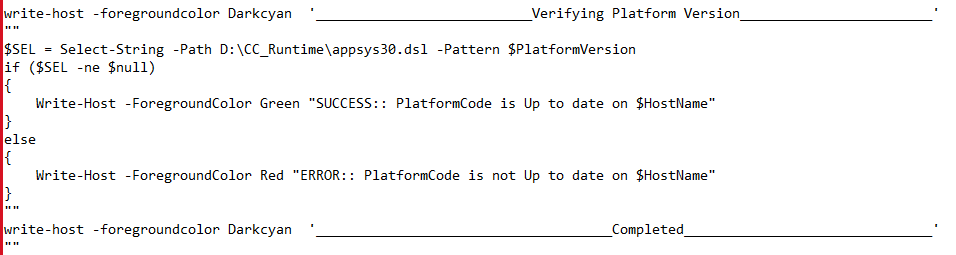
* In the below code the variable $Hostname = Hostname will take the server name to which the user is connected & It will verify the DSL version in about.dsl file for CoreIssue & CoreAuth

**-->**



* In the below code we can verify the platform version from appsys30.dsl file

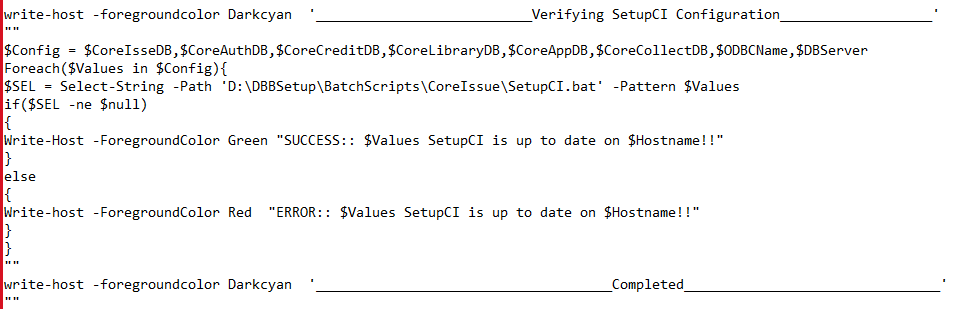
***-->***



* In the below code we can verify the SetupCI Configuration. All the variables declared in $Config will be compared with values present in SetupCI.bat file & Give the output accordingly to check if the SetupCI is upto date or not

Note - All values variables present in $Config are declared in SetupFile.ps1

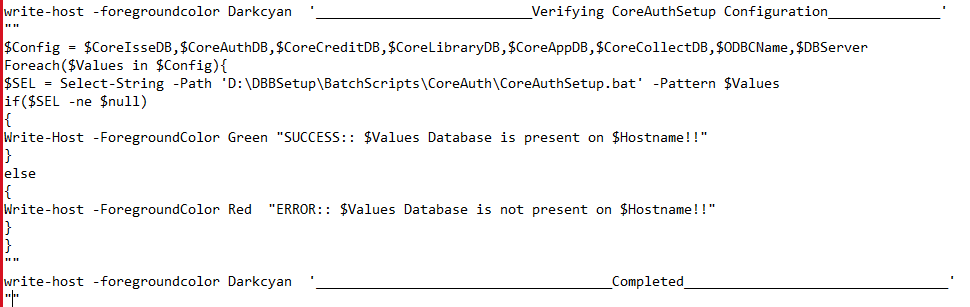
***-->***



* In the below code we can verify the CoreAuthSetup Configuration. All the variables declared in $Config will be compared with values present in CoreAuthSetup.bat file & Give the output accordingly to check if the CoreAuthSetup is upto date or not

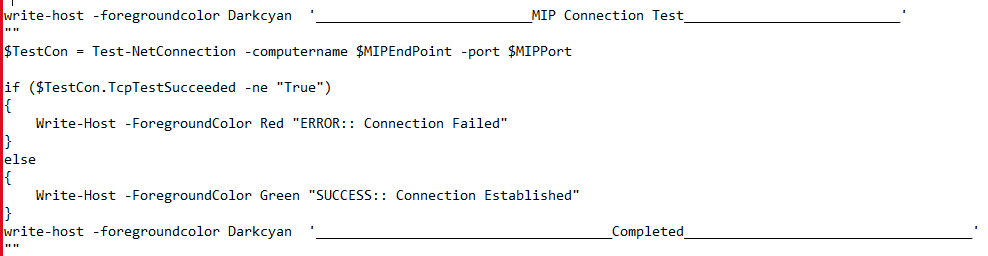
Note - All values variables present in $Config are declared in SetupFile.ps1

**-->**



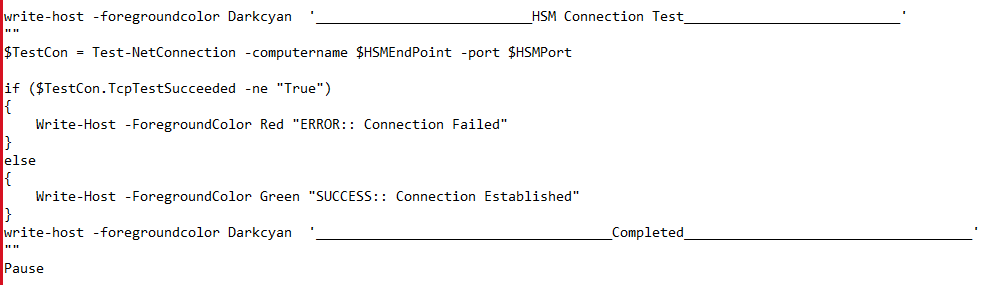
* In the below code If the connection is established on port $MIPport then the MIP Connection test is successful else it’s failed

**-->**



* In the below code If the connection is established on port $HSMport then the HSM Connection test is successful else it’s failed

**-->**



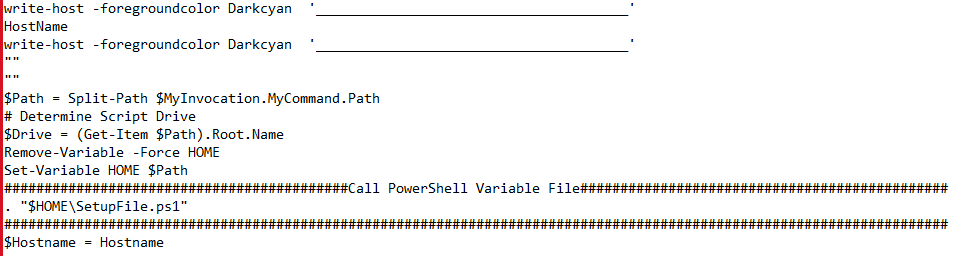
1. **WCF Validation Script explanation:**

****

* Below code is used to to keep your Script Dynamic. $MyInvocation will store the information of script, How it executed or How script got started/invoked.

$HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

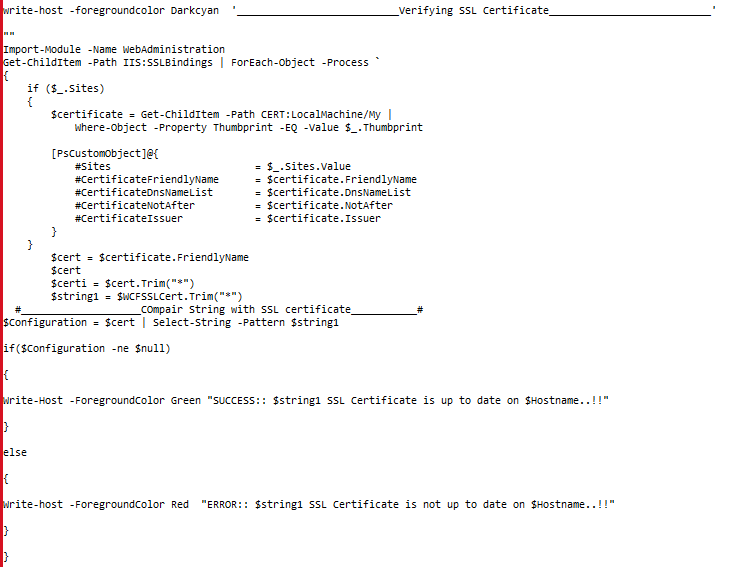
**-->**



* Below code is used to verify SSL certificate is up to date or not. In the variable $WCFSSLCert will contain the name of certificate.

It will compair the values of $WCFSSLCert with values of certificates present on all the sites & It will also print the friendly name of certificate of each site.

**-->**



* Below code is used to verify all the properties of WCF AppPool. First it will go into IIS:\Apppools\ and check for the WCF then it will verify all the properties mentioned below:

1. 32Bit Is True or False
2. Status of WCFAppPool
3. WebAppName
4. Version
5. State
6. UserIdentityType
7. Username

**-->**





* Below code is used to verify all the properties of CoreCARDServices AppPool. First it will go into IIS:\Apppools\ and check for the CoreCARDServices then it will verify all the properties mentioned below:

1. 32Bit Is True or False

2.Status of CoreCARDServices AppPool

3.WebAppName

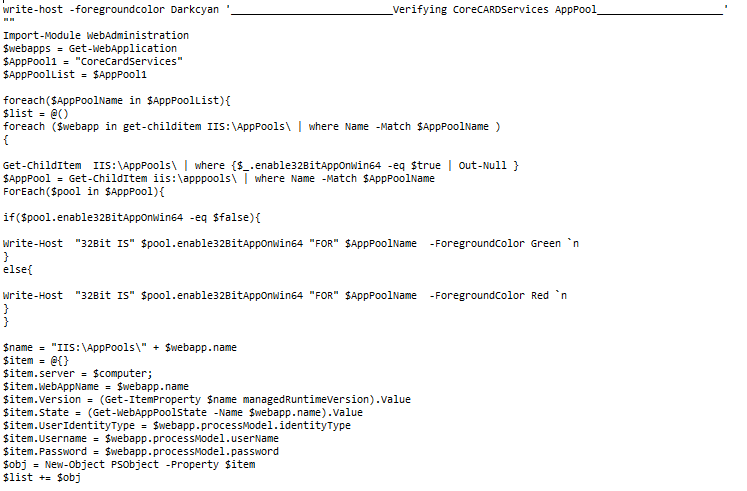
4. Version

5. State

6.UserIdentityType

7. Username

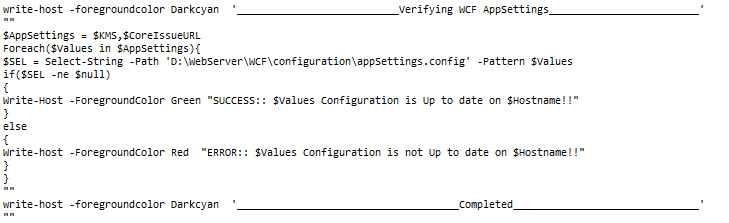
**-->**





* Below code is used to verify the $Values configuration is up to date for WCF AppSettings values $KMS & $CoreIssueURL

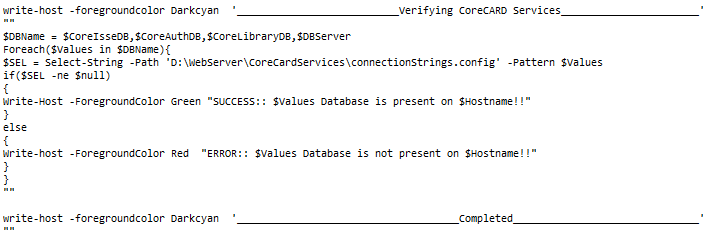
**-->**



* Below code is used to verify the $Values of $Database is up to date for CoreCARD Services.

It will check the $CoreIsseDB,$CoreAuthDB,$CoreLibraryDB,$DBServer is present or not on the server.

**-->**

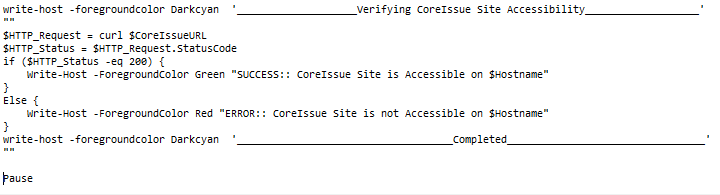


* Below code is used to verify the CoreIssue Site Accessibility.

$CoreIssueURL contains the url of CoreIssue.

CoreIssue Site will be accessible only when the HTTP Status of Site is equal to 200.

**-->**



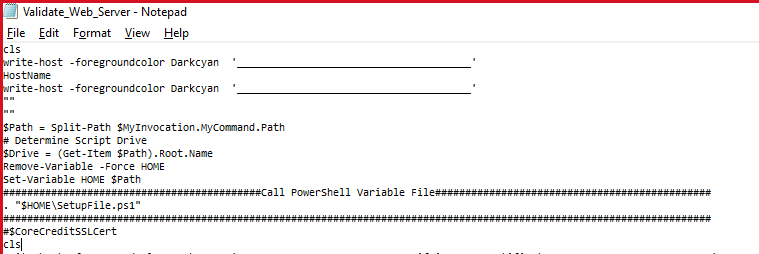
1. **WebServer Validation Script**

****

* Below code is used to to keep your Script Dynamic. $MyInvocation will store the information of script, How it executed or How script got started/invoked.

$HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

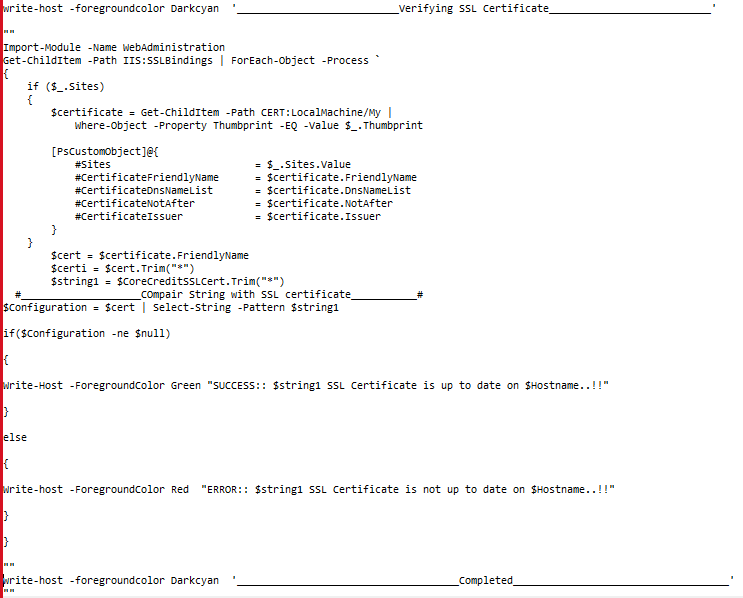
**-->**



* Below code is used to verify SSL certificate is up to date or not. In the variable $CoreCreditSSLCert will contain the name of certificate.

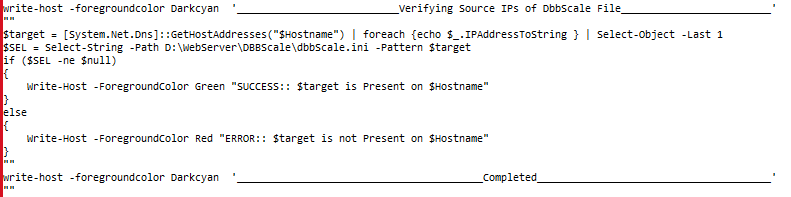
It will compair the values of $CoreCreditSSLCert with values of certificates present on all the sites & It will also print the friendly name of certificate of each site.

**-->**



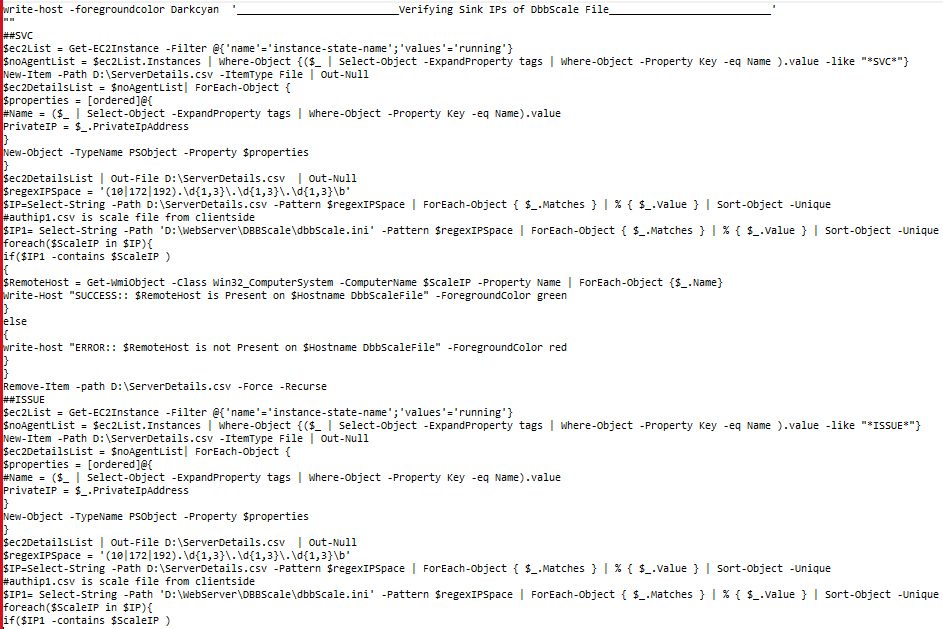
* Below code is used to take the Host IP and check it in file dbbscale.ini if it’s present or not

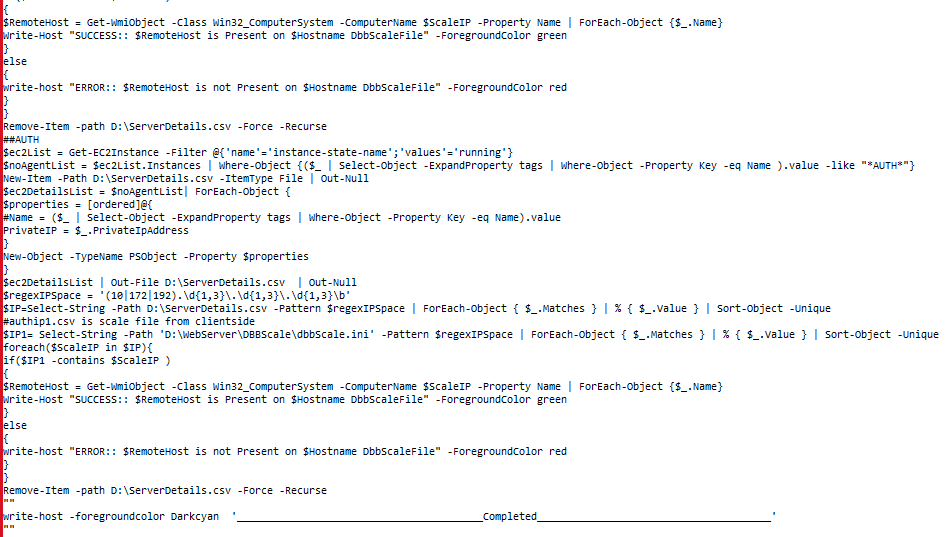
**-->**



* In below code the $ec2List.Instances will fetch the running instances of SVC, Issue & Auth. Then it will fetch the private IP addresses of these instances and it will store in file & then it will compare those IP addresses with Dbbscale.ini file and will check if IPs are present in this file or not in sink section of file

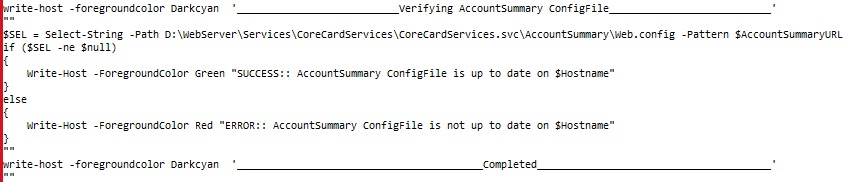
-->





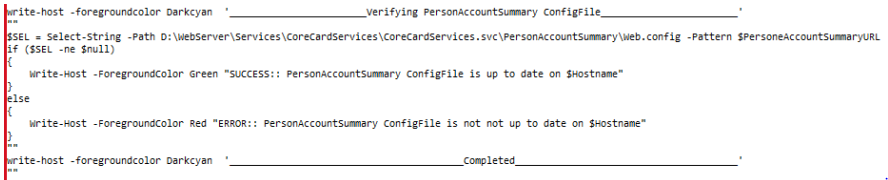
* Below code is used to check if $AccountSummaryURL is up to date or not in web.config file

**-->**



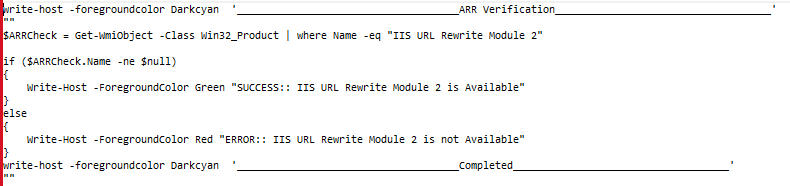
* Below code is used to check $PersonAccountSummaryURL is up to date or not in Web.config file

**-->**



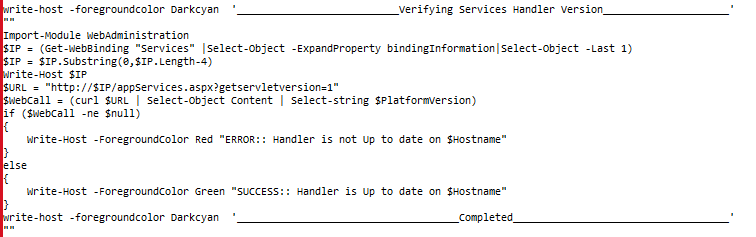
* Below code is used to check if the object “IIS URL rewrite Module 2” is available or not

**-->**



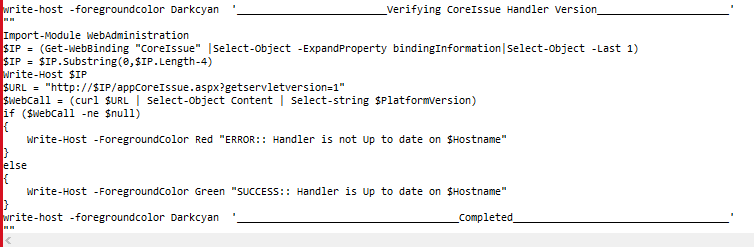
* In the Below code the first 3 lines will fetch the IP of **services** then it will add the IP in $URL along with serverletversion to check if handler version is up to date or not

**-->**



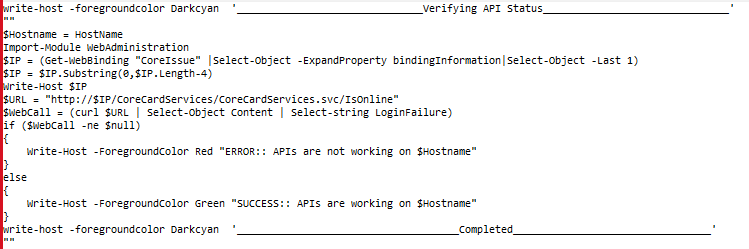
* In the Below code the first 3 lines will fetch the IP of **CoreIssue** then it will add the IP in $URL along with serverletversion to check if handler version is up to date or not

**-->**



* Following code will fetch the IP from Coreissue site. $URL will go inside the CoreCardServices/CoreCardServices.svc/IsOnline.  
  $WebCall will check the string IsOnline is present or not in CoreCardServices.svc. If string matches it will print API are working on $Hostname.

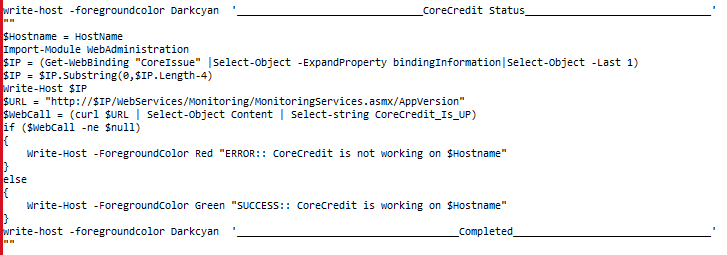
**-->**



* Following code will fetch the IP from Coreissue site. $URL will go inside the Webservices/Monitoring/MonitoringServices.asmx/Appversion

$WebCall will check the string CoreCredit is up or not in MonitoringServices.asmx. If string matches it will print CoreCredit is working on $Hostname.

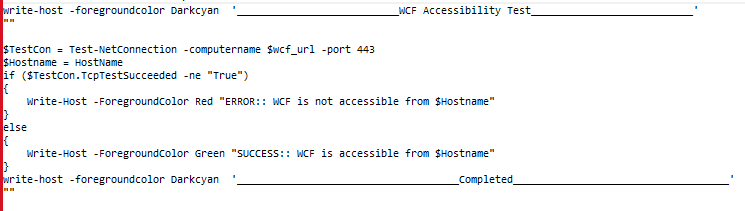
**-->**



* Below code is used to check the test Netconnection for WCF.

It will check if $wcf\_url is accessible or not from Host through port no. 443

-->



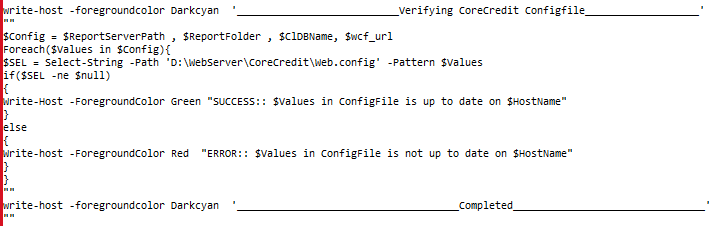
* Below code is used to verify the values in $Config are matching with CoreCredit\Web.Config file.

Values listed in $Config are as below:

1. $ReportServerPath
2. $ReportFolder
3. $ClDBName
4. $wcf\_url

If all values matches then it will print ConfigFile is up to date

-->



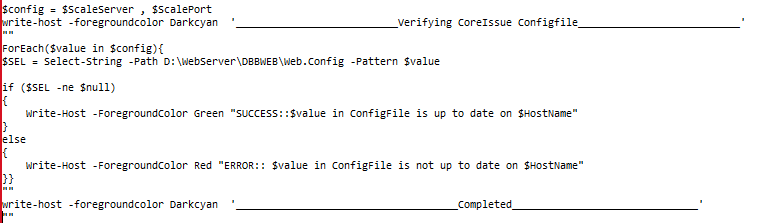
* Below code is used to verify the values in $Config are matching with DBBWEB\Web.Config file.

Values listed in $Config are as below:

1. $ScaleServer
2. $Scaleport

If all values matches then it will print ConfigFile is up to date

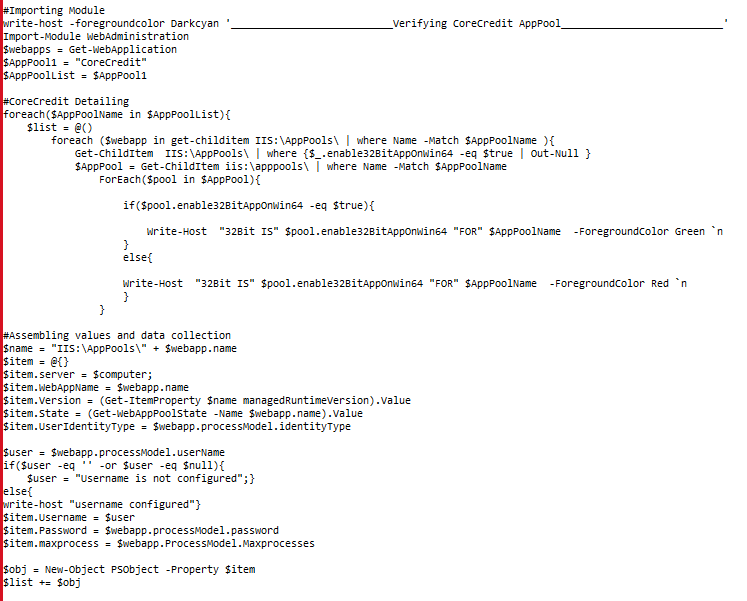
-->

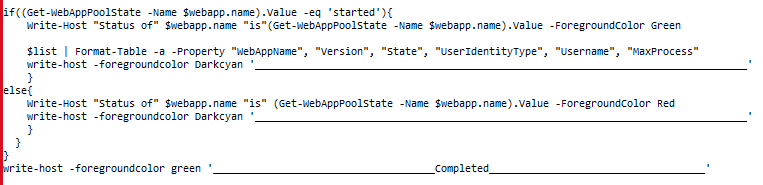


* Below code is used to verify all the properties of CoreCredit AppPool. First it will go into IIS:\Apppools\ and check for the CoreCredit then it will verify all the properties mentioned below:

1. 32Bit IS True or False
2. Status of CoreCredit AppPool
3. WebAppName
4. Version
5. State
6. UserIdentityType
7. Username
8. Maxprocess

-->

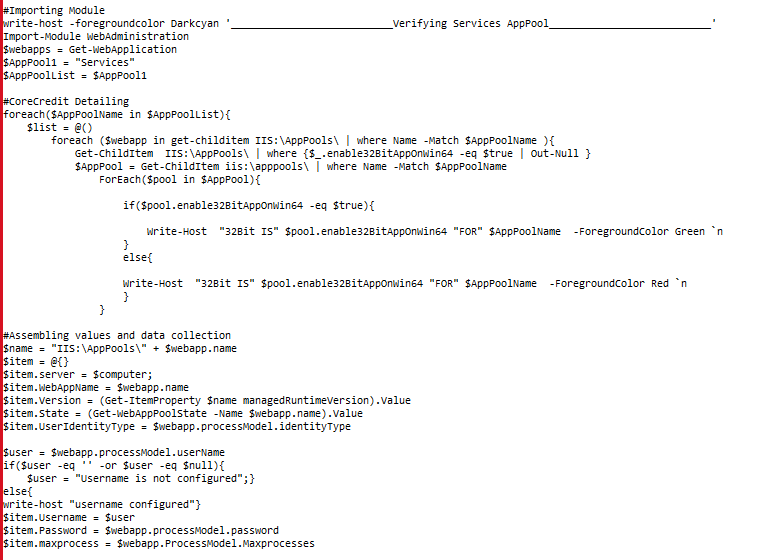


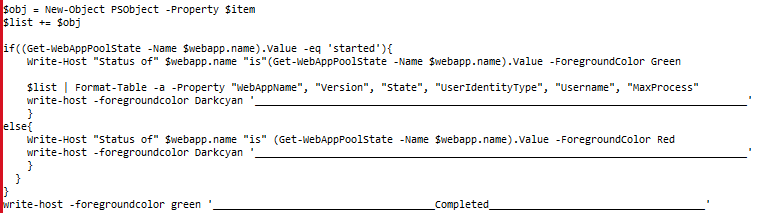


* Below code is used to verify all the properties of Services AppPool. First it will go into IIS:\AppPools\ and check for the Services then it will verify all the properties mentioned below:

1. 32Bit IS True or False
2. Status of Services AppPool
3. WebAppName
4. Version
5. State
6. UserIdentityType
7. Username
8. Maxprocess

-->





* Below code is used to verify all the properties of CoreIssue. First it will go into IIS:\AppPools\ and check for the CoreIssue then it will verify all the properties mentioned below:

1.32Bit IS True or False

2.Status of CoreIssue AppPool

3.WebAppName

4.Version

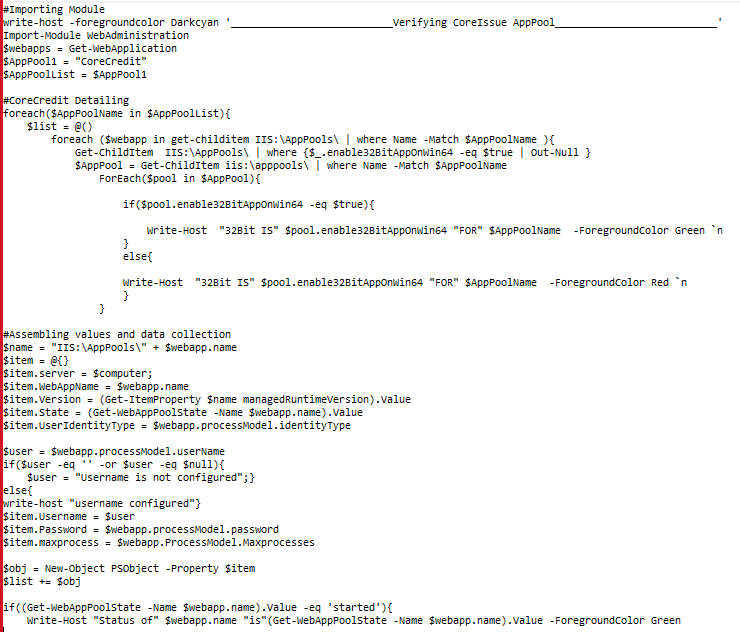
5.State

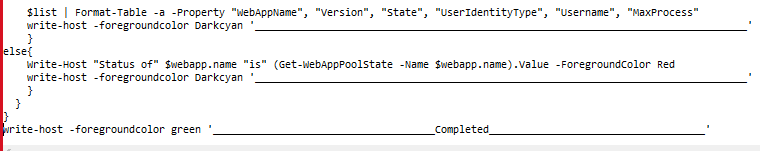
6.UserIdentityType

7.Username

8.Maxprocess

-->



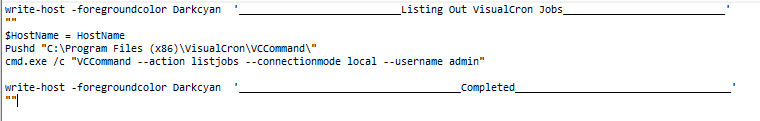


1. **VSCron Jobs Validation Script:**

****

* Below code is used to check All Visual Cron Jobs listed by username Admin by executing cmd.exe file

-->



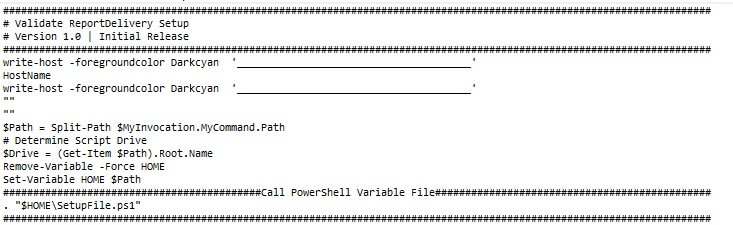
1. **RS Server Validation Script:**

****

* Below code is used to to keep your Script Dynamic. $MyInvocation will store the information of script, How it executed or How script got started/invoked.

$HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

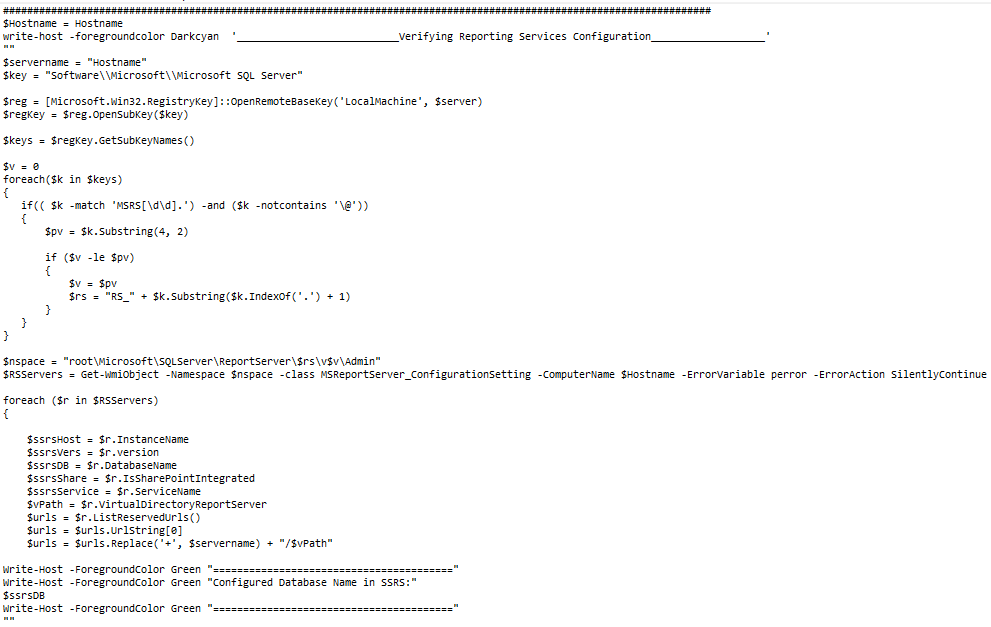
-->

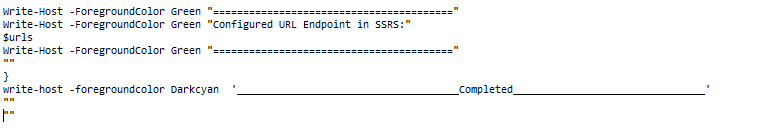


* Below code is used to verify reporting services configuration which is

fetching the keys from registry matching MSRS[\d\d] pattern and not containing \@ pattern. Also, Fetching WMI object from the sql root location "root\Microsoft\SQLServer\ReportServer\$rs\v$v\Admin"

and providing output of every object with there instance name and version and dbname

**-->** 



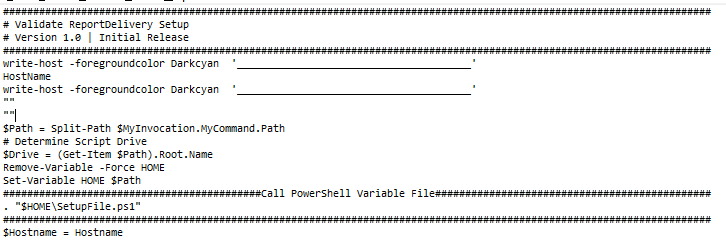
1. **RD Server Validation Script:**

****

* Below code is used to to keep your Script Dynamic. $MyInvocation will store the information of script, How it executed or How script got started/invoked.

$HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

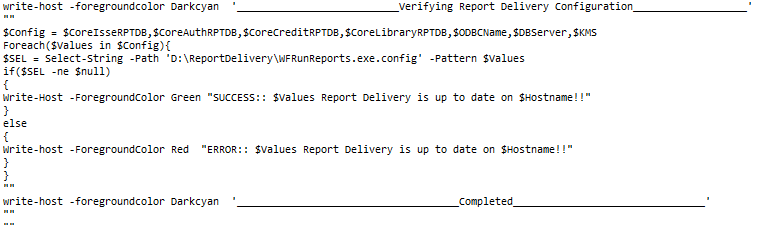
**-->**



* In the below code we can verify the values in WFRunReports.exe.config file. All the variables declared in $Config will be compared with values present in WFRunReports.exe.config file & Give the output accordingly to check if the Report Delivery is up to date or not

Note - All values of variables present in $Config are declared in SetupFile.ps1

**-->**



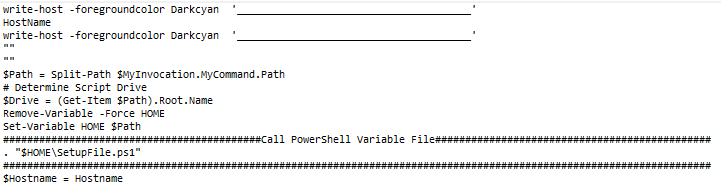
1. **KMS Server Validation Script:**

****

* Below code is used to to keep your Script Dynamic. $MyInvocation will store the information of script, How it executed or How script got started/invoked.

$HOME is a automatic variable that has read only access of users Home directory. It will help to call variable file i.e. SetUpFile.ps1

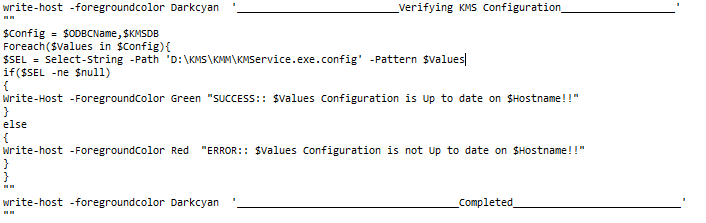
**-->**



* In the below code we can verify the KMS Configuration is up to date or not in file KMSService.exe.config

If the $Config values found in KMSService.exe.config file that means KMS Configuration is up to date.

**-->**



* In the below code we can verify the KMS Service is running or not.

If Status of $ServiceName is found as running the KMS Service is in running status Else it will found but not in running status. Else it will show Servicename not found.

--> 