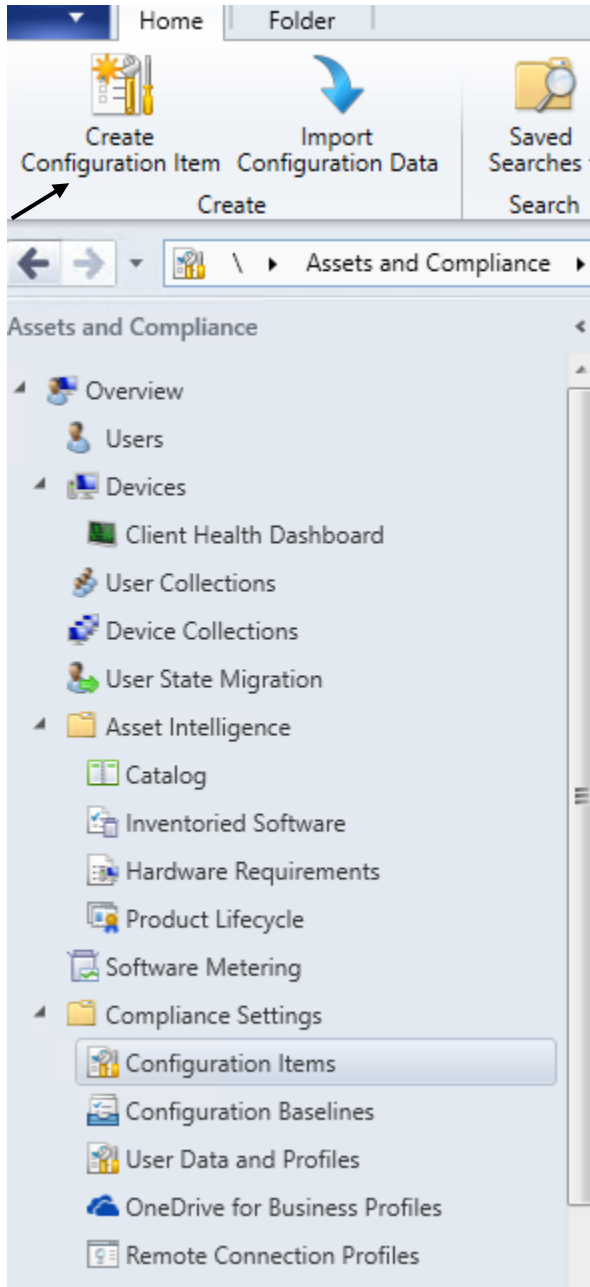


Office Activation Compliance Status

In this post, I will show you how to create Compliance Item and create a Base line compliance to check for Office activation status.





General

General

- Supported Platforms
- Settings
- Compliance Rules
- Summary
- Progress
- Completion

Specify general information about this configuration item

Configuration items define a configuration and associated validation criteria to be assessed for compliance on devices.

Name:

Description:

Specify the type of configuration item that you want to create:

Settings for devices managed with the Configuration Manager client

- Windows 10
- Mac OS X (custom)
- Windows Desktops and Servers (custom)
 - This configuration item contains application settings

Settings for devices managed without the Configuration Manager client

- Windows 8.1 and Windows 10
- Windows Phone
- iOS and Mac OS X
- Android and Samsung KNOX
- Android for Work

Assigned categories to improve searching and filtering:

Categories...

< Previous

Next >

Summary

Cancel



Supported Platforms

General

- Supported Platforms
- Settings
- Compliance Rules
- Summary
- Progress
- Completion

Specify the client operating systems that will assess this configuration item for compliance

Select the versions of Windows that will assess this configuration item for compliance:

Select all

- Windows XP
- Windows Vista
- Windows 7
- Windows 8
- Windows 8.1
- Windows 10
- Windows 2003
- Windows 2008
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Embedded
- Windows Server 2019

Specify the version of Windows manually:

Add...

< Previous

Next >

Summary

Cancel

Click Edit Script and paste the script as Windows PowerShell

The image shows two overlapping windows from the Windows Configuration Manager. The background window is 'Create Setting' with the 'Compliance Rules' tab selected. It shows a setting named 'Detect Office Activation' with a description field, 'Script' as the setting type, and 'String' as the data type. The 'Discovery script' section has an 'Edit Script...' button. The foreground window is 'Edit Discovery Script', which has a dropdown menu set to 'Windows PowerShell'. The script area contains PowerShell code for checking Office activation status across various versions (2007, 2010, 2013, 2016).


```
# Check for office 2007
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office12\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office12\OSPP.VBS" /dstatus
}
# Check for office 2010
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office12\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office12\OSPP.VBS" /dstatus
}
# Check for office 2013
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office15\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office15\OSPP.VBS" /dstatus
}
# Check for office 2016/office 365 proplus/office 2019
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office16\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office16\OSPP.VBS" /dstatus
}
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office16\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office16\OSPP.VBS" /dstatus
}
```

The image shows the 'Create Configuration Item Wizard' dialog box, 'Settings' tab. The left sidebar shows 'Settings' selected. The main area is titled 'Specify settings for this operating system' and contains a table of settings. The table has columns for Name, Setting Type, Inherited, and User Setting. One setting is listed: 'Detect Office Activation' with a 'Script' type, 'No' for Inherited, and 'No' for User Setting. At the bottom, there are 'New...', 'Edit...', and 'Delete' buttons, and a 'Next >' button highlighted.

Name	Setting Type	Inherited	User Setting
Detect Office Activation	Script	No	No

Click New

Create Configuration Item Wizard ✕

 Compliance Rules

General

Supported Platforms

Settings

Compliance Rules

Summary

Progress

Completion

Specify compliance rules for this operating system

Use compliance rules to specify the conditions that make a configuration item setting compliant on client devices. The following compliance rules are associated with this configuration item.

Name	Setting Name	CI Name	Condition	Severity	Remediate	R
There are no items to show in this view.						

Create Rule ✕

Specify rules to define compliance conditions for this setting

Name:

Description:

Selected setting:

Rule type:

The setting must comply with the following rule:

the following values:

Report noncompliance if this setting instance is not found

Noncompliance severity for reports:



General

Supported Platforms

Settings

Compliance Rules

Summary

Progress

Completion

Specify compliance rules for this operating system

Use compliance rules to specify the conditions that make a configuration item setting compliant on client devices. The following compliance rules are associated with this configuration item.

Name	Setting Name	CI Name	Condition	Severity	Remediate	R
Status	Detect Office Acti...	Office Activ...	Equals True	Information	No	

New... Edit... Delete

< Previous Next > Summary Cancel



General

Supported Platforms

Settings

Compliance Rules

Summary

Progress

Completion

The wizard will create an operating system configuration item with the following settings

Details:

The wizard will create an operating system configuration item with the following settings:

New operating system configuration item will be saved as:

- Name: Office Activation Status
- Description:
- Categories: "Baselines"

The following Windows versions are supported:

- All Windows 7 (64-bit)
- All Windows 7 (32-bit)
- Windows 7 (64-bit)
- Windows 7 SP1 (64-bit)
- Windows 7 (32-bit)
- Windows 7 SP1 (32-bit)
- All Windows 8 (64-bit)
- All Windows 8 (32-bit)
- All Windows 8.1 (64-bit)
- All Windows 8.1 (32-bit)
- All Windows 10 (64-bit)
- All Windows 10 (32-bit)
- All Windows Server 2016 and higher (64-bit)
- All Windows Server 2019 and higher (64-bit)

The following compliance rules are added:

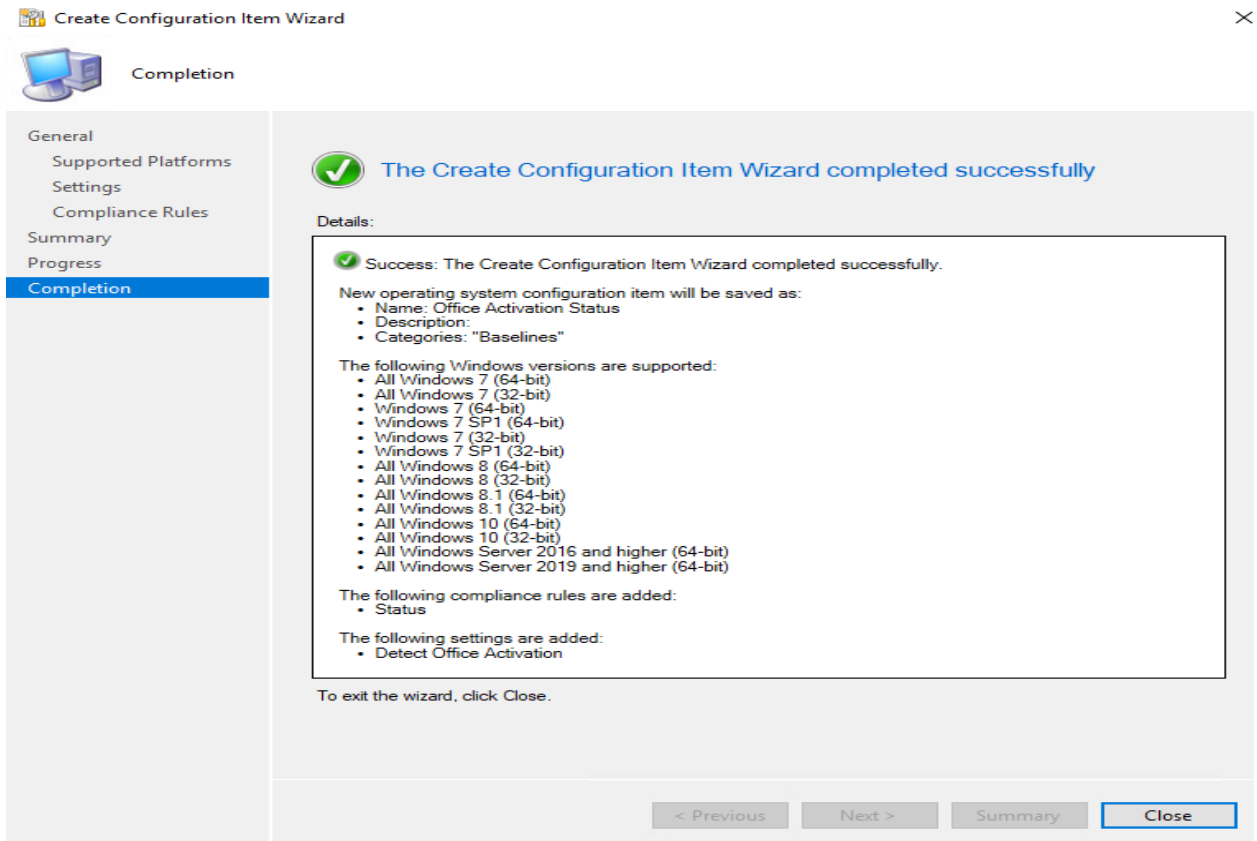
- Status

The following settings are added:

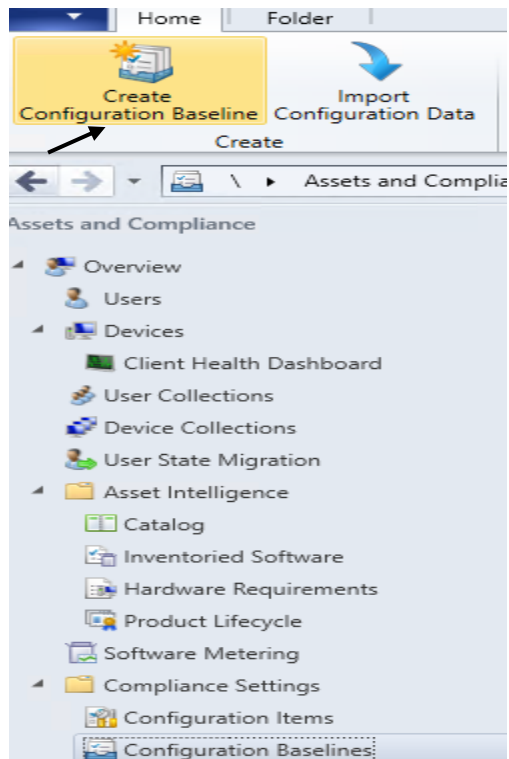
- Detect Office Activation

To change these settings, click Previous. To apply the settings, click Next.

< Previous Next > Summary Cancel



Now we have to create Configuration Baselines



Click Add and select Office Activation Status CI we just created.

Create Configuration Baseline

Specify general information about this configuration baseline

Name: CB - Office Activation

Description:

Select the configuration data (configuration items, configuration baselines, and software updates) to be evaluated for compliance by this configuration baseline. This configuration baseline will be assessed as compliant if all the items specified are compliant. Optional items are evaluated only if the relevant application is present on the client devices.

Configuration data:

Name	Type	Purpose	Revision
Office Activation Status	Operating System	Required	Latest

Always apply this baseline even for co-managed clients

Assigned categories to improve searching and filtering:

Now Deploy Configuration Baselines to a collection

Assets and Compliance > Overview > Compliance Settings > Configuration Baselines

Configuration Baselines 1 items

Icon	Name	Status	Deployed	User Setting	Date Modified	Compliance Count	Noncompliance Count	Failure Count	Modified By
	CB - Office Activation	Enabled	No	No	13-Sep-2019 9:57 AM	0	0	0	RAMLAN\Administrator

CB - Office Activation

Deploy Configuration Baselines

Select the configuration baselines that you want to deploy to a collection

Available configuration baselines:

Filter...

There are no items to show in this view.

Add >

< Remove

Selected configuration baselines:

Filter...

CB - Office Activation

Remediate noncompliant rules when supported

Allow remediation outside the maintenance window

Generate an alert:

When compliance is below: %

Date and time:

Generate System Center Operations Manager alert

Select the collection for this configuration baseline deployment.

Collection:

Schedule

Specify the compliance evaluation schedule for this configuration baseline:

Simple schedule

Run every:

Custom schedule

I am applying the CI/CB once only. No need for recurring schedule

Assets and Compliance > Overview > Compliance Settings > Configuration Baselines

Configuration Baselines 1 Items

Icon	Name	Status	Deployed	User Setting	Date Modified	Compliance Count	Noncompliance Count	Failure Count	Modified By
	CB - Office Activation	Enabled	Yes	No	13-Sep-2019 9:57 AM	0	0	0	RAMLAN/Administrator

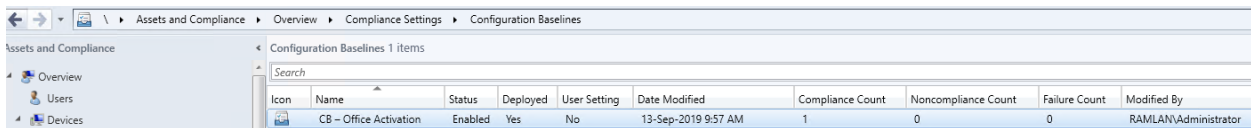
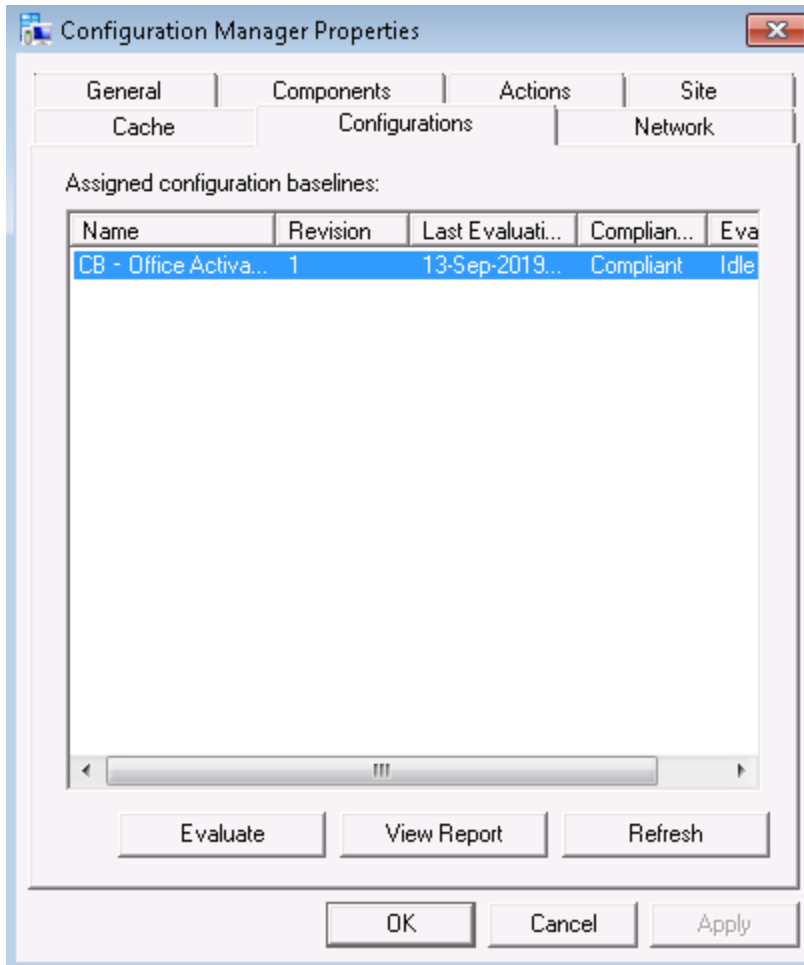
CB - Office Activation

Icon	Collection	Compliance %	Deployment Start Time	Action
	All Systems	0.0	15-Sep-2019 9:59 AM	Monitor

Summary | Deployments

Now sit back and relax for CI/CB to perform the action. Later we will check the status (Compliant or Non Compliant). If you want the policy to apply immediately – run machine policy.

Here is compliant status for one of the workstations, I checked.



This concludes Office Activation Status using Configuration Item and Configuration Baselines.

Thanks

Ram Lan
13th Sep 2019

DISCOVERY SCRIPT:

```
# Check for office 2007
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office12\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office12\OSPP.VBS"
/dstatus
}
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office12\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office12\OSPP.VBS"
/dstatus
}
# Check for office 2010
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office14\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office14\OSPP.VBS"
/dstatus
}
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office14\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office14\OSPP.VBS"
/dstatus
}

# Check for office 2013
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office15\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office15\OSPP.VBS"
/dstatus
}
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office15\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office15\OSPP.VBS"
/dstatus
}

# Check for office 2016/office 365 proplus/office 2019
if(Test-Path -Path "C:\Program Files (x86)\Microsoft Office\Office16\OSPP.VBS"){
    $status= cscript "C:\Program Files (x86)\Microsoft Office\Office16\OSPP.VBS"
/dstatus
}
if(Test-Path -Path "C:\Program Files\Microsoft Office\Office16\OSPP.VBS"){
    $status=cscript "C:\Program Files\Microsoft Office\Office16\OSPP.VBS"
/dstatus
}

if(( $status |Select-String -Pattern '---LICENSED---') -ne $null )
{
write-output "True"
}
else
{
write-output "False"
}
```