Defining a Policy

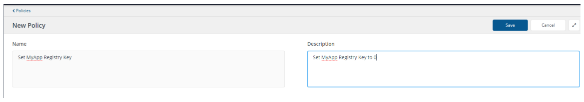
Summary

How to define a Policy in Guaranteed State.

On this page:

Defining a policy

You define a policy from the guaranteed state administration policies view. Enter a name and description for the policy and press **Save**.



Name: Set MyApp Registry Key

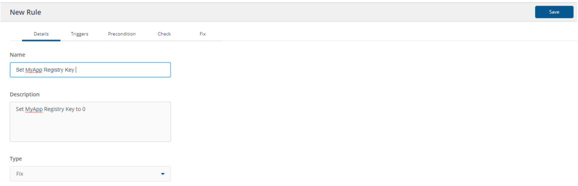
Description: Set MyApp Registry Key to 0

Managing rules

Policies need rules in order to be of any use. The following headings show how to define rules, caveats for editing rules and how to associate rules with policies.

Defining a rule

You define a rule from the guaranteed state administration rules view. Enter a name and description, and select a type (Check or Fix). In the example below, we are choosing a fix rule:



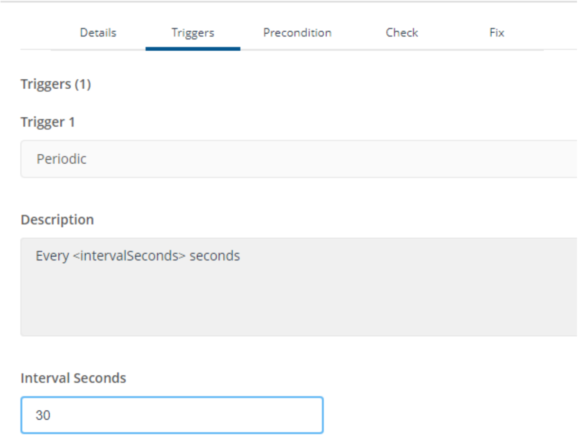
Name: Set MyApp Registry Key

Description: Set MyApp Registry Key to 0

Type: Fix

We are going to define that the rule should be evaluated periodically, every 30 seconds.

> Do not specify a period less than 30 seconds. The rule may be accepted by the management interface but the Tachyon Agent will not apply it.



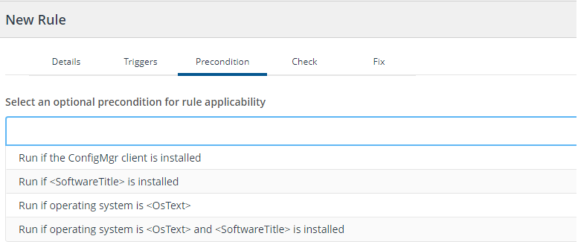
Triggers (1)

Trigger 1: Periodic

Description: Every <intervalSeconds> seconds

Interval Seconds: 30

We will not define a precondition, but you can see here the available choices, should you desire to set one.

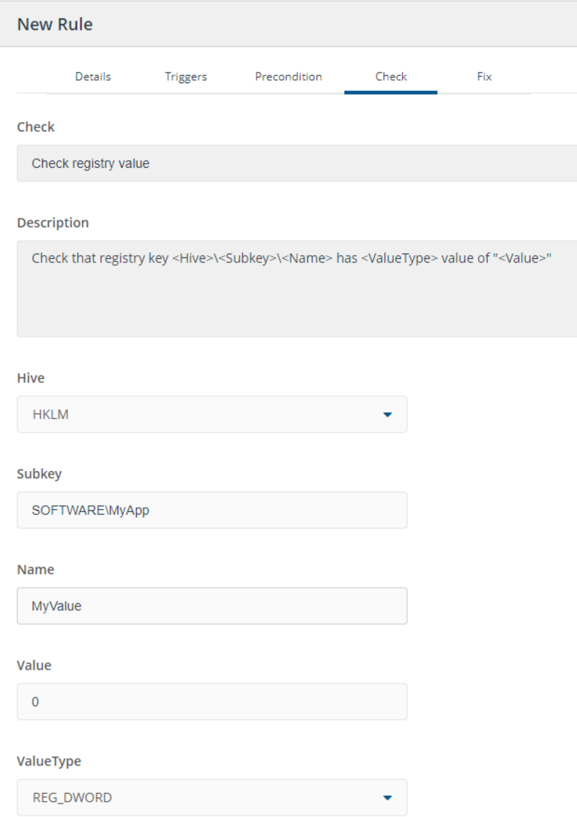


Select an optional precondition for rule applicability

- Run if the ConfigMgr client is installed
- Run if <SoftwareTitle> is installed
- Run if operating system is <OsText>
- Run if operating system is <OsText> and <SoftwareTitle> is installed

We will define a check rule that requires that there is a value **MyValue** in the registry hive HKLM and subkey **SOFTWARE\MyApp**. Its value should be of type **REG_DWORD** and it should be **0**.

If any part of the subkey is missing, or the value name or value fail to match this check rule, then the fix rule will be applied.



Check: Check registry value

Description: Check that registry key <Hive>\<Subkey>\<Name> has <ValueType> value of "<Value>"

Hive: HKLM

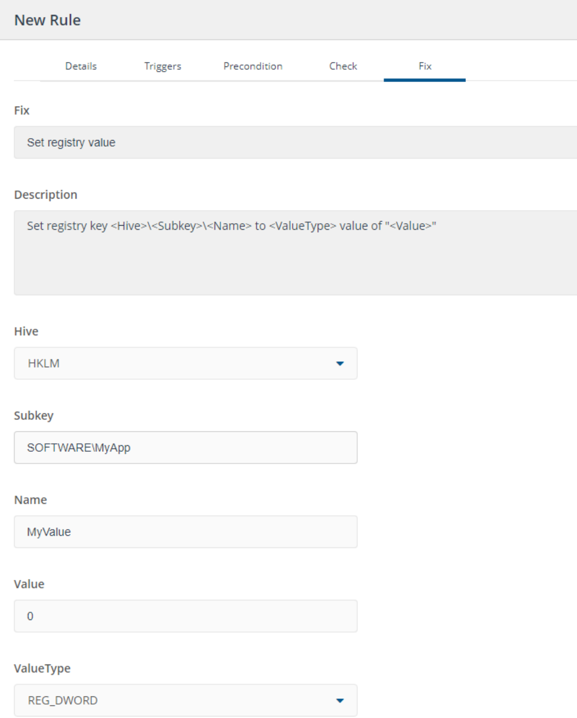
Subkey: SOFTWARE\MyApp

Name: MyValue

Value: 0

ValueType: REG_DWORD

The fix rule will be identical to the check rule. This means that if the registry is not consistent with the check rule, it will be made consistent. If any part of the subkey does not exist, or the value name is missing or has a different value or type, then all of these discrepancies will automatically be corrected and the registry value made to be consistent with the check rule.



Fix: Set registry value

Description: Set registry key <Hive>\<Subkey>\<Name> to <ValueType> value of "<Value>"

Hive: HKLM

Subkey: SOFTWARE\MyApp

Name: MyValue

Value: 0

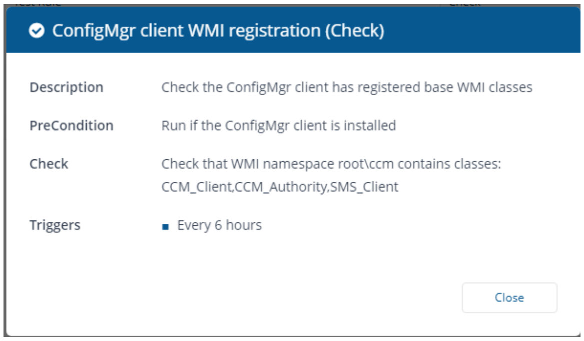
ValueType: REG_DWORD

Editing existing rules

> ⚠ In Guaranteed State for Tachyon 4.0 a rule cannot be edited once it has been saved. This is because a rule may have been associated with an active policy and changing it subsequently could have undesirable effects. However you can clone an existing rule and then edit the cloned rule to create a changed rule if you wish. To clone a rule, select it from the **Guaranteed StateAdministrationRules** page and then press **Clone**.

Viewing Rule Details

You can view the rule details for a rule by clicking the **Details** link displayed on the rules page. For example:



ConfigMgr client WMI registration (Check)

Description: Check the ConfigMgr client has registered base WMI classes

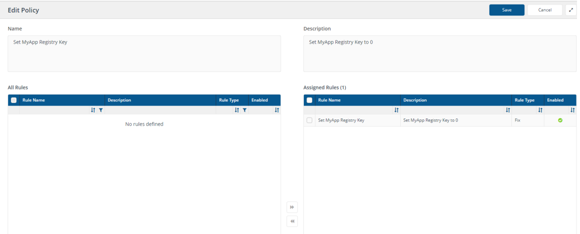
PreCondition: Run if the ConfigMgr client is installed

Check: Check that WMI namespace root\ccm contains classes: CCM_Client,CCM_Authority,SMS_Client

Triggers: ■ Every 6 hours

Associating a rule with a policy

We associate a rule with a policy by editing the policy. We see the available rules on the left of the page.



Edit Policy

Name: Set MyApp Registry Key

Description: Set MyApp Registry Key to 0

All Rules

Rule Name	Description	Rule Type	Enabled
No rules defined			

Assigned Rules (1)

Rule Name	Description	Rule Type	Enabled
Set MyApp Registry Key	Set MyApp Registry Key to 0	Fix	✓

We select the rule(s) we want to associate with the policy and then press the ⏩ button.

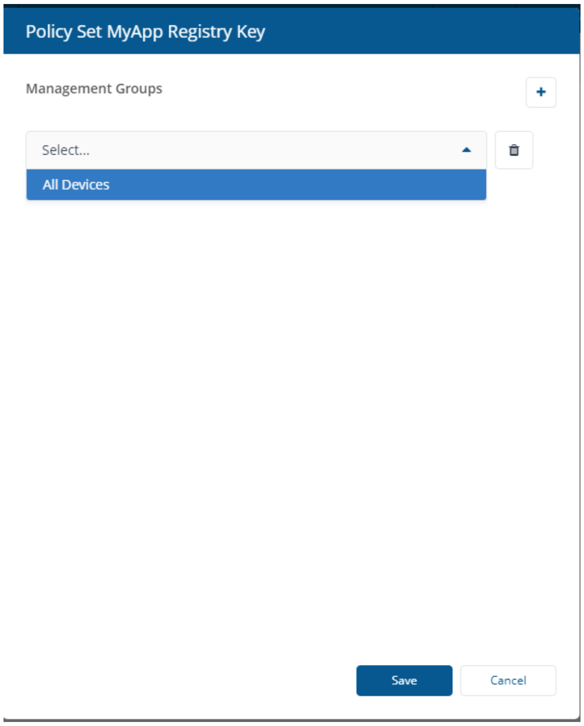
When we have selected all the rules we wish to associate, we press **Save:**

Assigning a policy to a management group

Before deploying the policy we must first define the devices to which the policy will be sent. We do this by assigning a policy to a management group.

To do this, select the policy and press the **Assign** button.

By default, there will always be a management group for **All Devices**. We select this and then press **Save**:



Policy Set MyApp Registry Key

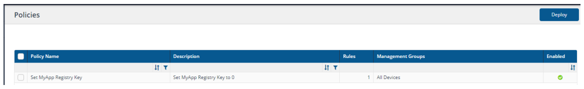
Management Groups

Select...

All Devices

We should now see the assigned policy.

⚠ The policy is now shown with its assigned management groups, in this case, the All Devices group.


Policies

Deploy

☐	Policy Name	Description	Rules	Management Groups	Enabled
☐	Set MyApp Registry Key	Set MyApp Registry Key to 0	1	All Devices	✓

Deploying policies

For a policy to become effective, it must be *deployed*. Policies are deployed by pressing the Deploy button from the policies view in the administration section of Guaranteed State.

Note that deployment affects **all visible policies** that are *enabled*. You can enable or disable a policy from this screen by selecting it and then pressing the **Disable** or **Enable** button (which will change caption as appropriate based on the current state of the policy).

Deleted policies and deployment

If a policy is deleted, then pressing Deploy will also cause any endpoint that receives the deployment to remove the policy and to cease to enforce it.

Disabled policies and deployment

If a policy is disabled, then pressing **Deploy** will also cause any endpoint that receives the deployment to remove the policy and to cease to enforce it.

If the policy is subsequently re-enabled, then the next time **Deploy** is pressed, it will be sent out again and become effective.

⚠ In this initial release of Guaranteed State, you are prompted to deploy a policy as soon as you create it. However, deploying a new policy where any of the following is true will not result in any action being taken at the endpoints:

- The policy is not associated with any rules
- The policy has not been associated with a management group
- All rules associated with a policy are disabled

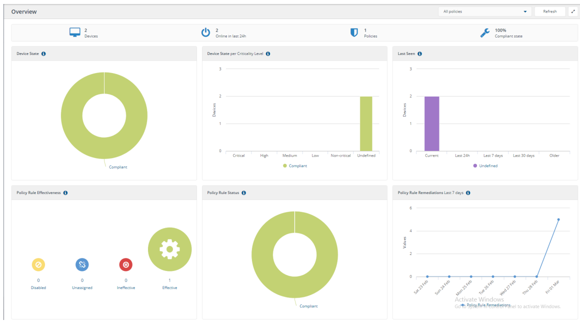
However, once a policy has been deployed, re-deploying the policy will always affect the endpoints to which the policy was previously deployed. In the case where a re-deployed policy meets any of the above criteria, it will cease to be effective at the endpoints

If the change removes a policy from any management groups, endpoints affected by this change WILL receive a policy update. For example, suppose you had a policy P1 assigned to management group MG1 and you then re-target P1 to management group MG2 instead and re-deploy it. All the endpoints which ever received policy P1 previously will have that policy removed if they now fall outside management group MG2.

The distribution of policies to endpoints is staggered. The stagger period associated with this activity is designed to avoid excessive network traffic. It is not related to the agent stagger parameter which is defined during agent installation.

Verifying policies

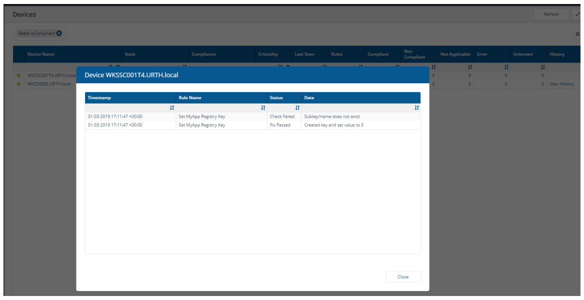
You can verify policy enforcement on the Overview page and confirm policy download and applications on the devices.



Verifying policy enforcement

The Guaranteed State overview shows us that we now have two compliant devices.

You can drill down on each chart. For example, clicking on the **Device State** c hart allows us to see the specific devices that are associated with the chart. From there we can examine the history of each device, for example.



Device WKSSC001T4.URTH.local

Timestamp	Rule Name	Status	Data
01-03-2019 17:11:47 +00:00	Set MyApp Registry Key	Check Failed	Subkey/name does not exist
01-03-2019 17:11:47 +00:00	Set MyApp Registry Key	Fix Passed	Created key and set value to 0

⚠ History entries are only created when the status changes state from pass to fail or vice-versa. Even though you may have a rule which evaluates at the endpoint periodically, a history entry is not created for each evaluation cycle. The endpoint only communicates a state change when the rule status changes.

This means that if the cause of a failure changes, you may not receive a history entry with the revised cause, because no updates are received until the compliance state changes. This limitation may be revised in a future release.

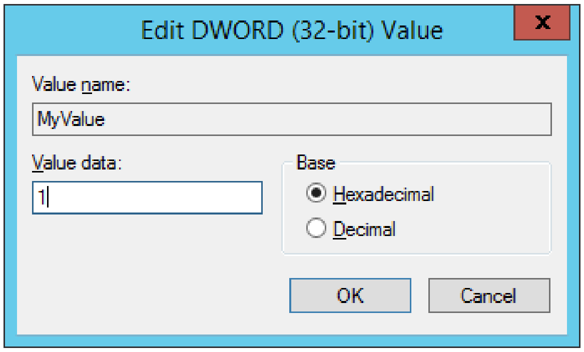
Verifying policy at the endpoint

We can also confirm that the policy was downloaded and applied at the endpoints by examining the Tachyon agent log on the endpoint. Here we see that the policy was downloaded and successfully applied

```
2019-03-01 17:11:47.268 [ 6892] INFO  - Downloaded and applied new policy with 1 rule(s) (829 bytes)
2019-03-01 17:11:47.271 [ 6892] INFO  - [Policy Rule 'Set MyApp Registry Key' (ID 4) (check)] Successfully processed instruction
2019-03-01 17:11:47.271 [ 6892] INFO  -  -> Policy Rule 'Set MyApp Registry Key' (ID 4) - check failed; running fix rule
2019-03-01 17:11:47.274 [ 6892] INFO  - [Policy Rule 'Set MyApp Registry Key' (ID 4) (fix)] Successfully processed instruction
2019-03-01 17:11:47.280 [ 6892] INFO  -  -> New Policy Rule 'Set MyApp Registry Key' (ID 4) evaluated; notifying status of 'fix passed'
```

If we now start the registry editor on the endpoint, we can see that the key and value exist.

If we attempt to change the registry key value to 1.

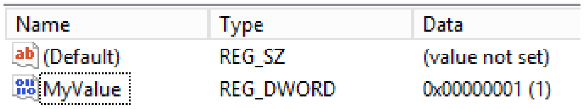

Edit DWORD (32-bit) Value

Value name: MyValue

Value data: 1

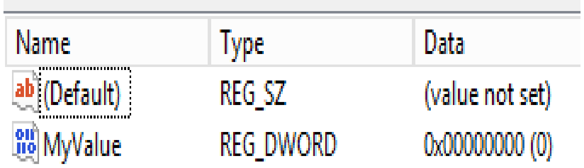
Base
● Hexadecimal
○ Decimal

We see that, temporarily, it is out of compliance with the check rule.



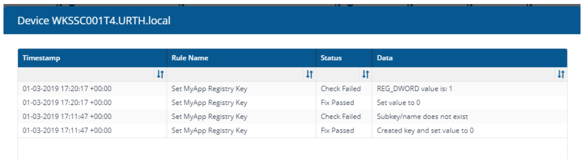
Name	Type	Data
(Default)	REG_SZ	(value not set)
MyValue	REG_DWORD	0x00000001 (1)

However, after 30 seconds, it is automatically reset to be compliant.



Name	Type	Data
(Default)	REG_SZ	(value not set)
MyValue	REG_DWORD	0x00000000 (0)

If we view the device history from the Guaranteed State Overview screen, we see that the momentary compliance violation has been logged.



Device WKSSC001T4.URTH.local

Timestamp	Rule Name	Status	Data
01-03-2019 17:20:17 +00:00	Set MyApp Registry Key	Check Failed	REG_DWORD value is: 1
01-03-2019 17:20:17 +00:00	Set MyApp Registry Key	Fix Passed	Set value to 0
01-03-2019 17:11:47 +00:00	Set MyApp Registry Key	Check Failed	Subkey/name does not exist
01-03-2019 17:11:47 +00:00	Set MyApp Registry Key	Fix Passed	Created key and set value to 0

Having done this you can now click the Explore button. This launches the standard Tachyon Explorer page but note that the device coverage for any question or action has been set automatically.

Please refer to Using Explorer to investigate devices in Guaranteed State.