

# Règles d'application des modèles par rapport aux champs collectés du serveur VMWare

## Sommaire

[Qu'est-ce qu'une règle d'application des modèles ?](#)  
[Définition d'une règle](#)  
[Exemple](#)  
[Les Commentaires](#)  
[Syntaxe d'une condition d'une règle d'application](#)  
[Syntaxes pour les champs des éléments de la source contenant des listes](#)  
[Surcharge des règles par défaut](#)  
[Chemin du fichier de définition des règles d'application](#)  
[Visualiser les règles d'application \( Onglet "Règles d'application des modèles" \)](#)

## Qu'est-ce qu'une règle d'application des modèles ?



Une règle d'application des modèles permet de définir une ou plusieurs condition(s), permettant à la source d'attribuer un modèle à un hôte en fonction de la valeur d'un champ de l'API VMWare.

La source dispose de règles d'application fournies par défaut. Il vous est possible de :

- créer vos propres règles d'application,
- de désactiver les règles par défaut.

## Définition d'une règle

Une règle d'application est écrite au format JSON ( *clé* <-> *valeur* ):

clé	commentaire
<code>name</code>	Nom de la règle <div style="border: 1px solid red; padding: 5px; margin-top: 10px;"> Doit être unique dans le JSON</div>
<code>template</code>	Nom du ou des modèle(s) qui seront attachés. Les modèles peuvent être listés en les séparant par une virgule.
<code>conditionX</code>	Défini une condition d'application de la règle : <ul style="list-style-type: none"><li>• X étant le numéro de la condition.</li><li>• Les conditions sont interprétées dans l'ordre.</li><li>• Entre chaque condition, il s'agit d'un OU logique, ce qui signifie qu'il suffit qu'une condition soit validé pour appliquer la règle.</li></ul>
<code>disable</code>	<ul style="list-style-type: none"><li>• Deux valeurs possibles ( <i>Par défaut, la valeur est false</i> ) :<ul style="list-style-type: none"><li>◦ <b>true</b> : la règle est désactivée, le modèle défini dans la règle ne sera pas appliqué.</li><li>◦ <b>false</b> : la règle est activée.</li></ul></li></ul> <p>Non obligatoire</p> <p> La valeur de la clé ne prend pas en compte les majuscules, <b>false</b> et <b>False</b> sont pareils.</p>

## Exemple

```
[
  {
    "name" : "rule 32 bit generation",
    "template" : "x86,32-bit",
    "condition1" : "config.product.osType=86",
    "disable" : "true"
  },
  {
    "name" : "windows Server 64bits",
    "template" : "windows_server,64-bit,x64",
    "condition1" : "config.product.osType=64 AND config.guestFullName=windows server",
    "condition2" : "config.product.osType=64 AND config.guestFullName=^Microsoft Server$",
    "condition3" : "config.product.osType=64 AND config.guestFullName=vista server"
  }
]
```

Dans l'exemple, nous pouvons voir qu'il y a deux règles qui sont définies.

- Ici la première règle nommée "**rule 32 bit generation**" possède une condition sur le type du système, si cette condition est respectée alors les modèles "**x86**" et "**32-bit**" seront appliqués à l'élément créé via la source.
- La seconde règle nommée "**windows Server 64bits**", fonctionne de la même manière que la première, mais avec plus de conditions.

## Les Commentaires

Il est possible d'ajouter des commentaires dans ce fichier.

Toutes les lignes qui commencent par le caractère # seront considérées comme des commentaires.

## Syntaxe d'une condition d'une règle d'application

Une condition est séparée en 2 parties et utilise le caractère "=" comme symbole séparateur :

- à gauche, le champ de l'API VMWare qui va être inspecté ;
- à droite, la condition sur la valeur du champ de l'API à remplir ;



Cette valeur ne prend pas en compte les majuscules, ce qui fait que *WINDOWS* et *windows* sont pareils

### Exemple: os=myOS

Dans cet exemple pour que les modèles soient ajoutés, il faut que le nom de l'OS (système d'exploitation) contienne le texte "myOS".

Voici les syntaxes possibles pour les conditions des règles d'application des modèles :

Signification	Syntaxe	Description	Exemple
Contient	<i>NOM_DE_CHAMP</i> = <i>texte</i>	La condition signifie que le champ VMWare choisi <b>doit CONTENIR</b> le texte.	os=myOS
Commence par	<i>NOM_DE_CHAMP</i> = ^ <i>texte</i>	Si l'expression commence par le caractère "^", la condition signifie que le résultat attendu <b>doit COMMENCER</b> par le texte	macvendor=^myMacV endor
Termine par	<i>NOM_DE_CHAMP</i> = <i>texte</i> \$	Si l'expression termine par le caractère "\$" la condition signifie que le résultat attendu <b>doit TERMINER</b> par le texte.	ostype=myType\$
Est strictement égal	<i>NOM_DE_CHAMP</i> = ^ <i>texte</i> \$	Si l'expression commence par "^" ET termine par "\$", la condition signifie que le résultat attendu <b>doit être l'expression EXACTE</b>	osversion=^2.6.0\$
Une condition et l'autre sont remplies	<i>Condition1</i> AND <i>Condition2</i>	Le modèle sera appliqué si la condition 1 et 2 sont remplies	os=myOS AND osversion=^2



### Remarque

Note : c'est le 1er '=' qui sépare le nom du champ et le texte.

**Exemple: config.guestFullName=linux=(Ubuntu)**

config.guestFullName est le nom du champ VMWare

linux=(Ubuntu) est le texte recherché

## Syntaxes pour les champs des éléments de la source contenant des listes

La syntaxe de VALUES est la même que celle des règles de mapping.

( Voir la page [Mapping des informations collectées des champs du serveur VMWare vers les propriétés et les données Shinken](#) )

Signification	Syntaxe	Description	Exemple
Contient	<b>NOM_DE_CHAMP&gt;VALUES(CONDITION)=texte</b>  ( le champ doit être de type liste )	Ici texte doit correspondre au résultat de l'application de la règle VALUES et de sa condition telles que décrites dans la page sur les règles de mapping.	shinken.tags>VALUES(=^bdx)=bdx_prod,bdx_preprod  shinken.tags_by_category.OS>VALUES(0)=Linux

## Surcharge des règles par défaut

Les règles par défaut peuvent être surchargées en créant une règle avec le même nom.

### Exemple:

Le modèle ajouté pour la règle *set template for virtual machines* sera maintenant **vmware virtual host**



Noter qu'une surcharge ne possédant pas au minimum un nom et un modèle ne sera pas prise en compte

```
[
  {
    "name" : "set template for virtual machines",
    "template" : "vmware virtual host"
  }
]
```

Il est possible de désactiver les règles par défaut en créant une règle avec le même nom et la clé **désable** à **true**.

( dans ce cas, le modèle n'est pas nécessaire )

```
[
  {
    "name" : "set template for virtual machines",
    "disable" : "true"
  }
]
```

Lors d'une surcharge, si les clés **template** ou **condition** ne sont pas spécifiés, les valeurs par défaut de celle-ci sont récupérées si la règle est activée.

Exemple avec la règle *set template for virtual machines* définies à droite :

```
[
  {
    "name" : "set template for virtual machines",
    "template" : "vmware virtual host",
    "disable" : "false"
  }
]
```

## Avant surcharge de la règle *set template for virtual machines*

Configuration						
Règles d'application des modèles						
Mapping vers les propriétés et les données de Shinken						
Résumé des dernières exécutions						
Détail de la dernière exécution						
Fichier de configuration <code>/etc/shinken-user/source-data/source-data-Mon-Collecteur-Synchronizer-VMWare/configuration/host_template_binding/host_template_binding_rule.json</code>						
Prob.	Défini par	Numéro	Nom de la règle	Conditions	Modèles	
--	--	--	--	--	--	
	défait	1	set template for esx	(1) <code>shinken.machine_type=HOST</code>	esx	
	défait	2	set template for virtual machines	(1) <code>shinken.machine_type=VIRTUAL_MACHINE</code>	virtual machine	
	défait	3	set template windows for virtual machines	(1) <code>config.guestFullName=Windows AND shinken.machine_type=VIRTUAL_MACHINE</code>	windows	
	défait	4	set template linux for virtual machines	(1) <code>config.guestFullName=Linux AND shinken.machine_type=VIRTUAL_MACHINE</code>	linux	

### Après surcharge de la règle *set template for virtual machines*

Configuration						
Règles d'application des modèles						
Mapping vers les propriétés et les données de Shinken						
Résumé des dernières exécutions						
Détail de la dernière exécution						
Fichier de configuration <code>/etc/shinken-user/source-data/source-data-Mon-Collecteur-Synchronizer-VMWare/configuration/host_template_binding/host_template_binding_rule.json</code>						
Prob.	Défini par	Numéro	Nom de la règle	Conditions	Modèles	
--	--	--	--	--	--	
	l'utilisateur	1	set template for virtual machines	(1) <code>shinken.machine_type=VIRTUAL_MACHINE</code>	virtual machine host	
	défait	2	set template for esx	(1) <code>shinken.machine_type=HOST</code>	esx	
	défait	3	set template windows for virtual machines	(1) <code>config.guestFullName=Windows AND shinken.machine_type=VIRTUAL_MACHINE</code>	windows	
	défait	4	set template linux for virtual machines	(1) <code>config.guestFullName=Linux AND shinken.machine_type=VIRTUAL_MACHINE</code>	linux	


### Chemin du fichier de définition des règles d'application

Pour ajouter une règle utilisateur d'application des modèles, il faut éditer le chemin suivant :

```
/etc/shinken-user/source-data/source-data-[nom de la source]/configuration/host_template_binding/host_template_binding_rule.json
```

#### Exemple de chemin pour la source livrée par défaut :

```
/etc/shinken-user/source-data/source-data-synchronizer-collector-vmware/configuration/host_template_binding/host_template_binding_rule.json
```

 Retrouvez la liste complète des champs collectés par la source sur la page [Liste des champs collectés auprès des VCenters ou ESXis](#)

Visualiser les règles d'application ( Onglet "Règles d'application des modèles" )

Pour visualiser la liste des règles définies pour cette source, rendez-vous dans l'onglet "**Règles d'application des modèles**".

Dans cet onglet, vous trouverez le chemin du fichier de définition des règles d'application des modèles utilisateur ( **1** ).

- ⚠ Un message d'avertissement apparaîtra si le fichier n'existe pas. Pour résoudre ce problème, créer le fichier avec le chemin indiqué sur l'interface puis appuyer le bouton de rafraîchissement ( **2** ). Si le fichier de définition est vide alors les valeurs par défaut seront utilisées.
- ⚠ Si une règle est désactivée, les erreurs sur là cette règle ne seront pas affichées.

Dans l'en-tête, vous trouverez aussi le bouton de rafraîchissement de la liste des règles ( **2** ).

- Ce bouton permet de rafraîchir cette liste sans devoir redémarrer le Synchronizer.
- Il faudra relancer un import pour réappliquer les nouvelles règles sur les éléments importés.

La liste des règles ( **3** ) comporte à la fois les règles définies par l'utilisateur et les règles par défaut.  
La liste affiche pour chaque règle :

- Le type de la règle ( *Utilisateur, par défaut ou désactivé par l'utilisateur* )
- Le numéro qui correspond à l'ordre d'application de la règle.
- Le nom de la règle
- La condition
- Les modèles qui seront appliqués

The screenshot shows the 'Règles d'application des modèles' tab in a software interface. At the top, there are navigation tabs: 'Configuration', 'Règles d'application des modèles' (selected), 'Mapping vers les propriétés et les données de Shinken', 'Résumé des dernières exécutions', and 'Détail de la dernière exécution'. Below the tabs, there is a 'Fichier de configuration' field with a refresh button (2) and a path: '/etc/shinken-user/source-data/source-data-Mon-Collecteur-Synchronizer-VMWare/configuration/host\_template\_binding/host\_template\_binding\_rule.json'. A table lists rules with columns: 'Prob.', 'Défini par', 'Numéro', 'Nom de la règle', 'Conditions', and 'Modèles'. The table contains six rows. The first row is highlighted in blue and has a red circle (3) next to its name. The last row has a red circle (4) next to its name and an 'F1 Aide' button. The interface also features a search bar and filter dropdowns at the top of the table.

Prob.	Défini par	Numéro	Nom de la règle	Conditions	Modèles
--	--	--	--	--	--
	utilisateur	1	windows server 64 bits	(1) config.product.osType=64 AND config.guestF.ullName=windows server (2) config.product.osType=64 AND config.guestF.ullName=*Microsoft Servers (3) config.product.osType=64 AND config.guestF.ullName=vista server	windows_server 64-bit x64
	utilisateur	2	rule 32 bit generation	(1) config.product.osType=86	x86 32-bit
	défaut	3	set template for esx	(1) shinken.machine_type=HOST	esx
	défaut	4	set template for virtual machines	(1) shinken.machine_type=VIRTUAL_MACHINE	virtual machine
	défaut	5	set template windows for virtual machines	(1) config.guestFullName=Windows AND shinke.n.machine_type=VIRTUAL_MACHINE	windows
	défaut	6	set template linux for virtual machines	(1) config.guestFullName=Linux AND shinken.ma.chine_type=VIRTUAL_MACHINE	linux

En bas à droite ( **4** ) se trouve le bouton pour afficher l'aide de la page.

Cette page d'aide explique le fonctionnement de l'onglet et peut être ouverte en appuyant sur la touche **F1**.

En utilisant la syntaxe de définition des règles, vous pourrez définir vos propres règles ou surcharger celles existantes par défaut.

- Les règles utilisateur apparaîtront en **bleu** dans la liste.
- Les règles désactivées apparaîtront en **gris** ( **5** ).

**Avant**

Configuration		Règles d'application des modèles	Mapping vers les propriétés et les données de Shinken	Résumé des dernières exécutions	Détail de la dernière exécution
Fichier de configuration <code>/etc/shinken-user/source-data/source-data-Mon-Collecteur-Synchronizer-VMWare/configuration/host_template_binding/host_template_binding_rule.json</code>					
Prob.	Défini par	Numéro	Nom de la règle	Conditions	Modèles
--	--	--	--	--	--
	l'utilisateur	1	windows server 64 bits	(1) <code>config.product.osType=64 AND config.guestF ullName=windows server</code> (2) <code>config.product.osType=64 AND config.guestF ullName="Microsoft Server\$</code> (3) <code>config.product.osType=64 AND config.guestF ullName=vista server</code>	windows_server 64-bit x64
	l'utilisateur	2	rule 32 bit generation	(1) <code>config.product.osType=86</code>	x86 32-bit
	défaut	3	set template for esx	(1) <code>shinken.machine_type=HOST</code>	esx
	défaut	4	set template for virtual machines	(1) <code>shinken.machine_type=VIRTUAL_MACHINE</code>	virtual machine
	défaut	5	set template windows for virtual machines	(1) <code>config.guestFullName=Windows AND shinke n.machine_type=VIRTUAL_MACHINE</code>	windows
	défaut	6	set template linux for virtual machines	(1) <code>config.guestFullName=Linux AND shinken.ma chine_type=VIRTUAL_MACHINE</code>	linux

## Après

Sources > Collecteur > Mon-Collecteur-Synchronizer-VMWare **Erreur** Impossible de récupérer les éléments depuis le serveur vSphere, vCenter ou ESXi distant 195.154.79.33. Le serveur

Configuration		Règles d'application des modèles	Mapping vers les propriétés et les données de Shinken	Résumé des dernières exécutions	Détail de la dernière exécution
Fichier de configuration <code>/etc/shinken-user/source-data/source-data-Mon-Collecteur-Synchronizer-VMWare/configuration/host_template_binding/host_template_binding_rule.json</code>					
Prob.	Défini par	Numéro	Nom de la règle	Conditions	Modèles
--	--	--	--	--	--
	l'utilisateur	1	windows server 64 bits	(1) <code>config.product.osType=64 AND config.guestF ullName=windows server</code> (2) <code>config.product.osType=64 AND config.guestF ullName="Microsoft Server\$</code> (3) <code>config.product.osType=64 AND config.guestF ullName=vista server</code>	windows_server 64-bit x64
	défaut	2	set template for esx	(1) <code>shinken.machine_type=HOST</code>	esx
	défaut	3	set template for virtual machines	(1) <code>shinken.machine_type=VIRTUAL_MACHINE</code>	virtual machine
	défaut	4	set template windows for virtual machines	(1) <code>config.guestFullName=Windows AND shinke n.machine_type=VIRTUAL_MACHINE</code>	windows
	défaut	5	set template linux for virtual machines	(1) <code>config.guestFullName=Linux AND shinken.ma chine_type=VIRTUAL_MACHINE</code>	linux
<b>5</b>	l'utilisateur-désactivé	6	rule 32 bit generation	(1) <code>config.product.osType=86</code>	x86 32-bit

Si un champ obligatoire est manquant ou erroné, un **Avertissement** ou une **Erreur** apparaîtra sur la règle en question.

