

Tagger sur le nom de l'hôte

Sommaire

Fonctionnement

Comment définir une règle Regexp

Cas d'exemple: automatiquement assigner au royaume Bordeaux les hôtes/clusters dont le nom commence par bdx

Exemple 1: mettre la valeur du royaume directement dans la valeur du royaume

Exemple 2: plus flexible, rajouter un modèle "bordeaux" (prioritaire sur les autres modèles)

Visualisation dans l'interface de configuration

Fonctionnement

Vous pouvez définir un tagger qui va utiliser automatiquement le nom pour modifier des propriétés sur les hôtes / clusters qui respectent sa règle de nommage (*une regexp*) sur une propriété.

Typiquement, le cas d'usage classique est de rajouter/modifier des templates d'hôtes/clusters sur les éléments s'ils respectent une règle de nommage sur le nom.

Comment définir une règle Regexp

Pour chaque règle Regexp, vous devrez définir un nouveau module et l'ajouter dans la configuration du tagger regexp-tags .

Vous pouvez copier l'exemple `/etc/shinken/modules/sync-regexp-tag.cfg` dans un nouveau fichier pour le modifier

```
define module{
  module_name      sync-regexp-tag-windows
  module_type      sync-regexp-tag
  matched_prop     host_name
  matched_regexp   .*srvwin.*
  method           append
  property         use
  value            windows
}
```

Les propriétés sont :

- **module_name**: le nom du module (*doit être unique*)
- **module_type**: doit être égal à sync-regexp-tag
- **matched_prop**: la propriété que l'on va regarder pour vérifier qu'elle correspond à ce que l'on recherche.
 - Par défaut, la propriété est le **host_name** (*Nom de l'objet hôte/cluster*)
- **matched_regexp**: Regexp auquel la propriété doit correspondre
- **method**: Comment la modification de la propriété va avoir lieu sur l'élément :
 - **replace**: si aucune valeur n'était définie dans la propriété à vérifier, le contenu de value sera mis dans la propriété visée, c.a.d. property.
 - **append**: ajoute le contenu de value à la fin dans la propriété visée, c.a.d. property.
 - **prepend**: ajoute le contenu de value au début de la propriété visée, c.a.d. property.
 - **set**: force le contenu de value dans la propriété visée, c.a.d. property.
- **property**: quelle propriété modifier
- **value**: la liste des modèles qui seront ajoutés.

Vous devez alors éditer le fichier rexexp-tags pour lier le nouveau tagger dans le fichier `/etc/shinken/taggers/regexp-tags.cfg`:

```
define tagger {
  tagger_name      regexp-tags
  order            1
  modules          sync-regexp-tag,sync-regexp-tag-windows
  description      This tagger will tag host based on the host_name
}
```

i La propriété `tagger_name` ne doit pas contenir les caractères suivants:

- <
- >
- =
- '

Une fois votre fichier sauvegardé, vous devez l'ajouter dans la liste des taggers du Synchronizer concerné (*par exemple /etc/shinken/synchronizers/synchronizer-master.cfg*)

```
define synchronizer {
    [ ... ]

    # Taggers:
    # ip-tags
    # regexp-tags
    taggers          ip-tags, regexp-tags, my-new-tagger

    [ ... ]
}
```

Pour que les modifications soient prises en compte, vous devez ensuite redémarrer le Synchronizer.

Si votre configuration est correcte, vous devriez retrouver votre tagger en bas de page de l'interface de configuration :

The screenshot shows the Shinken configuration interface. At the top, there are three synchronizer entries:

- 29 Désactivé **synchronizer-celle** (with a sub-entry **ctor-vmware-florie** and a note "Erreur de configuration")
- 30 Actif **discovery** (with a note "Prêt à être importé")
- 33 Actif **syncui** (with a note "Calcul des différences")

Below this, there is a "Taggers >" button and a table listing the configured taggers:

Ordre	Nom	Module
1	ip-tags	ip-tag-dmz
1	regexp-tags	sync-regexp-tag
1	my-new-tagger	

A red arrow points to the **my-new-tagger** entry in the table.

Cas d'exemple: automatiquement assigner au royaume Bordeaux les hôtes/clusters dont le nom commence par **bdx**

Exemple 1: mettre la valeur du royaume directement dans la valeur du royaume

Si nous avons des hôtes ou clusters dont le nom commence par la chaîne **bdx** nous savons qu'il est situé à Bordeaux. Il est donc intéressant de le assigner automatiquement au royaume **Bordeaux**, ce qui permettra de les superviser au plus proche possible (et donc avoir les bonnes règles de firewall par exemple pour lancer les sondes vers les serveurs).

Nous pouvons donc définir le module suivant que nous accrochons au tagger (comme indiqué précédemment).

Il va fonctionner ainsi:

- **SI** la propriété `host_name` (*le nom*) de l'hôte/cluster *commence par bdx*
 - **ALORS** on va écraser (`method=set`) la propriété `realm` (`property`) avec la valeur **Bordeaux** (`value`).

```

define module{
    module_name      sync-regex-tag-bordeaux-basic
    module_type      sync-regex-tag

    # La regexp a appliquer
    matched_regex    ^bdx.*

    # on va appliquer la regexp sur le nom de l'hote
    /cluster
    matched_prop     host_name

    # On va alors ecraser la propriete realm avec la
    valeur Bordeaux
    property         realm
    method           set
    value            Bordeaux
}

```

Exemple 2: plus flexible, rajouter un modèle "bordeaux" (prioritaire sur les autres modèles)

Si la méthode 1 fonctionne, elle n'est pas optimale: en effet, changer le royaume par Bordeaux est utile, mais dans le futur la localisation à Bordeaux va peu être demander à ce qu'une équipe locale ait les droits d'accès et de notifications sur la machine par exemple.

Il faudrait alors faire un second module, ce n'est pas l'idéal.

Il est donc **fortement recommandé** de ne pas modifier les propriétés directement, mais plutôt de **passer par des modèles d'hôtes/cluster**.

Il sera ainsi facile de faire des changements en masse sur toutes les machines Bordeaux, en modifiant juste le modèle Bordeaux.

Nous pouvons donc définir le module suivant que nous accrochons au tagger (*comme indiqué au dessus*).

Il va:

- **SI** le propriété *host_name* (*le nom*) de l'hôte/cluster *commence par bdx*
 - **ALORS** on va **rajouter au début** (*method=prepend*) de la propriété *use* (*property*) avec la valeur **bordeaux** (*value*).

```

define module{
    module_name      sync-regex-tag-bordeaux-
    recomanded
    module_type      sync-regex-tag

    # La regexp a appliquer
    matched_regex    ^bdx.*

    # on va appliquer la regexp sur le nom de l'hote
    /cluster
    matched_prop     host_name

    # On va alors rajouter le template bordeaux au
    debut de use (prioritaire)
    property         use
    method           prepend
    value            bordeaux
}

```

Visualisation dans l'interface de configuration

Vous pouvez consulter la configuration des différents taggers sur nom d'hôtes présent sur le Synchronizer sur la page d'accueil en cliquant sur **regex-tags**.

Clé	Valeur
Ordre	1
Modules	sync-regex-tag
Description	This tagger will tag host based on the host_name

Un onglet vous permet de visualiser également un résumé des règles qui vont s'appliquer.

Taggers > regex-tags

Configuration

Règles

Propriété ciblée

Expression Régulière

Propriété

Valeur

Méthode

host_name

.*shinidemo.*

use

SHINZEN-DEMO-REGEXP-EXAMPLE

append