Définir de nouveaux royaumes

Sommaire

Multi clients et/ou datacenters: ROYAUMES

L'architecture de Shinken Enterprise comme vue précédemment permet d'avoir un emplacement unique pour la configuration (*la politique de supervision et la définition des démons*).

- Les hôtes sont répartis parmi les Schedulers, qui définissent les commandes à exécuter.
- Les Pollers viennent alors récupérer ces tâches pour les exécuter.

En fait, si vous avez une architecture distribuée sur plusieurs continents, vous pouvez avoir des problèmes avec une vision si simple. Si l'architecture est commune à plusieurs réseaux, un Scheduler d'un client A peut avoir un Poller d'un client B lui demandant des tâches, ce qui n'est pas une bonne idée pour des questions d'efficacité du réseau (*même avec un réseau distribué*).

C'est là que devient intéressante la gestion des clients par site. Dans Shinken Enterprise, on le gère à travers les "royaumes".

Un royaume est un groupe de démons Shinken qui va gérer les hôtes/clusters:

- Un hôte/cluster ne peut pas appartenir à plusieurs royaumes.

Un royaume :

- peut avoir un Scheduler (*obligatoire si un hôte est défini dans ce royaume*)
- peut avoir un Poller (*facultatif si un royaume parent possède un Poller avec l'option manage_sub_realms à 1*)
- peut avoir un Reactionner (*facultatif si un royaume parent possède un Reactionner avec l'option manage_sub_realms à 1*)
- peut avoir un Broker (*facultatif si un royaume parent possède un Broker avec l'option manage_sub_realms à 1*)
- peut avoir un Receiver.

Dans un royaume, tous les Pollers vont prendre les tâches de tous les Schedulers de ce royaume.

Important

Il n'y a qu'**UN SEUL Arbiter** (*et son spare*) pour **TOUS** les royaumes. L'Arbiter gère TOUS les royaumes et ce qui s'y rattache.

Sous-royaumes

Un royaume peut avoir des sous-royaumes.

- Cela ne change rien pour les Schedulers, mais peut être utile pour les autres satellites et spares.
- Les Reactionners et Brokers sont liés au même royaume, mais ils peuvent traiter les tâches des sous-royaumes également. De cette façon, vous pouvez avoir moins de Reactionners et de Brokers.

Cela se fait grâce au paramètre **manage_sub_realms:**

- 0 (*valeur par défaut des Pollers et reactionners*) => le démon ne prendra du travail que dans son royaume, pas dans les sous-royaumes
- 1 (*valeur par défaut des Brokers et receivers*) > le démon va récupérer du travail dans son royaume et ses sous royaumes

Exemples d'utilisation des royaumes

Pour faire simple : vous mettez vos hôtes dans un royaume.

- Celui-ci est considéré comme un pool de ressources.
- Vous n'avez pas besoin de modifier la définition de vos hôtes/clusters si vous avez besoin de plus/moins de performance dans le royaume ou si vous souhaitez rajouter des satellites.

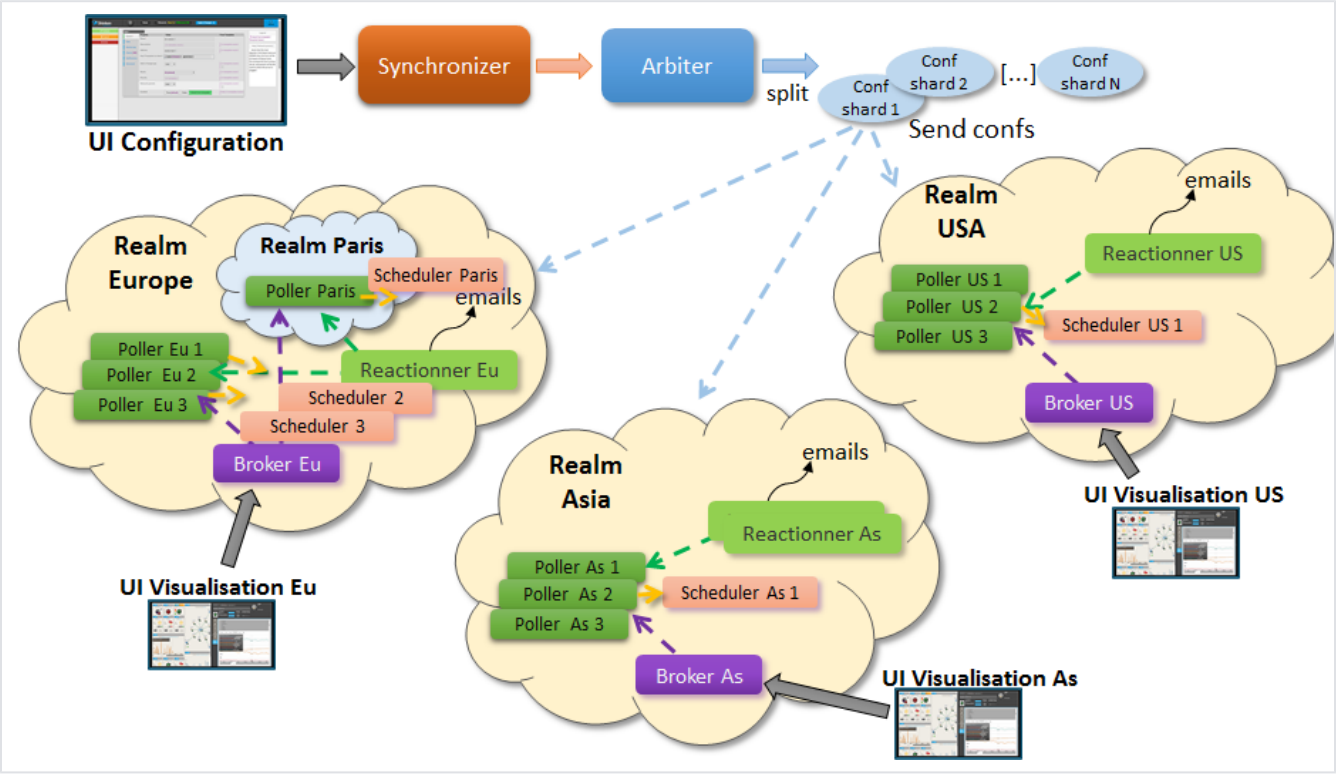
Les royaumes sont une manière de gérer les ressources. Ce sont plusieurs petits "nuages" dans votre infrastructure cloud globale.

PS : cette fonctionnalité est optionnelle, un royaume par défaut est créé (*son nom est **All***).

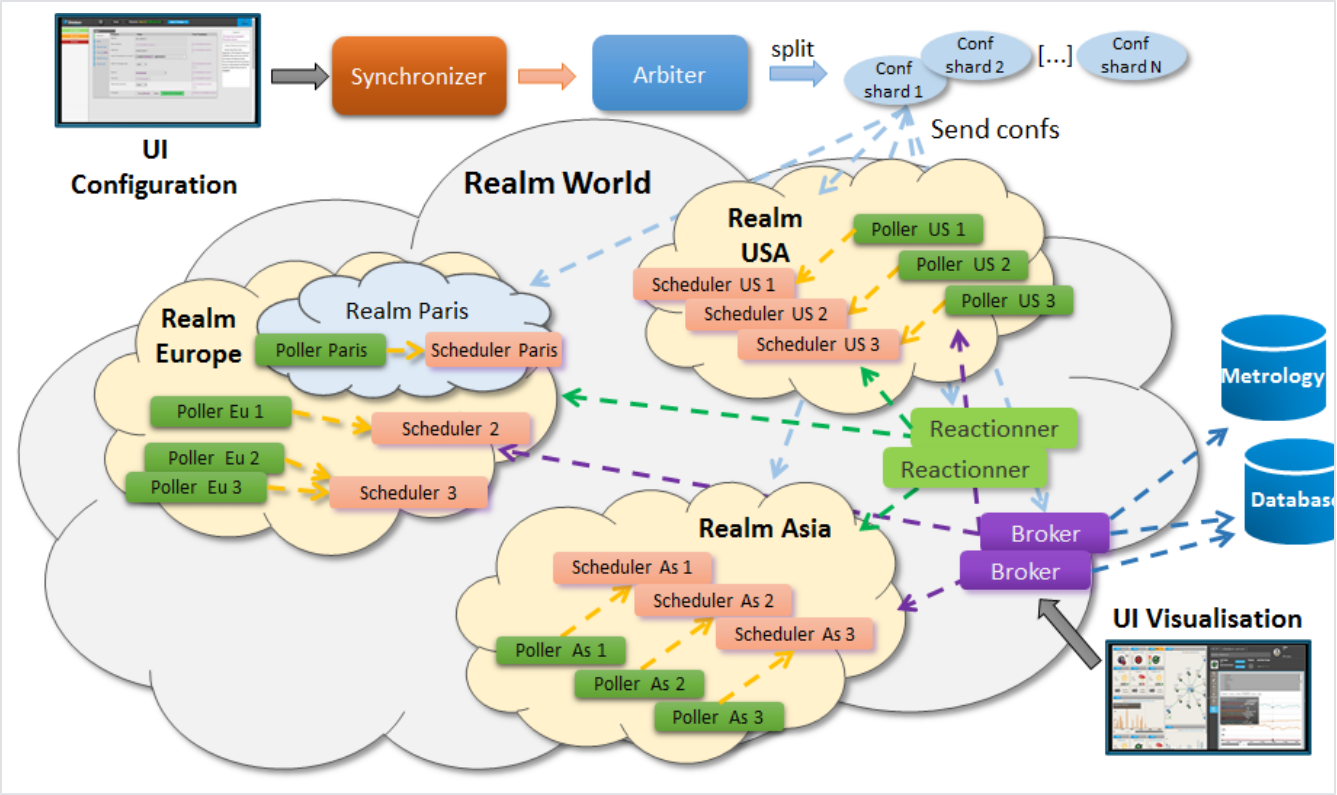
Prenons 2 exemples d'architecture distribuée à travers le monde.

- Dans le premier cas, l'administrateur ne souhaite pas partager les ressources entre royaumes.
- Dans le second, les Reactionners et Brokers sont partagés entre royaumes (*donc toutes les notifications sont envoyées d'un seul endroit, idem pour le stockage des données*).

Version isolée (cas 1)



Version partagée (cas 2)



Comme vous le voyez, tous les éléments sont dans un royaume unique, on utilise le sous-royaume pour les Reactionners/Brokers.

Royaume Monde et ses sous-Royaumes (Configuration)

Dans le répertoire **/etc/shinken/realms/**, voici la configuration pour l'architecture distribuée avec des royaumes :

Realm

```
define realm {
    realm_name      World
    # Now you define SUB REALMS of World
    realm_members   Europe,US,Asia
    # Element without explicit realm setting will be set in the World realm
    default         1
}

# We define our SUB REALMS
# EUROPE
define realm {
    realm_name       Europe
    # This one have it's own SUB REALM
    realm_members    Paris
}
# Paris: sub realm for Europe
define realm {
    realm_name       Paris
}

# USA
define realm {
    realm_name       USA
}

# Asia
define realm {
    realm_name       Asia
}


# For example the daemons for the Paris realm
define scheduler {
    scheduler_name     scheduler_Paris
    realm              Paris
}

# Example of a TOP level realm (WORLD) daemon that can reach daemons of the SUB realms
# so will reach Europe, Paris, USA and Asia
define reactionner {
    reactionner_name     reactionner-master
    manage_sub_realms    1
    realm                World
}
```

La propriété realm_name ne doit pas contenir les caractères suivants:

- <
- >
- "
- '

Pour une gestion plus facile des royaumes, nous conseillons de placer chaque définition de royaumes dans un fichier dédié (*royaume All dans all.cfg, royaume Paris dans paris.cfg, etc...*).

De la même manière, nous conseillons aussi fortement de placer les définitions des démons dans le dossier approprié puisque les outils livrés avec Shinken supposent que ces conventions sont respectées.

Ainsi, on placera par exemple les définitions des Reactionners dans /etc/shinken/reactionners, des Brokers dans /etc/shinken/brokers etc...

Pour attacher un hôte ou un cluster à un royaume, il faut :

- Aller dans la page de configuration de l'élément
- Sélectionner le royaume souhaité dans le champ "Royaume"