

Modèle proxmox-ve-by-API__lifecycle

Sommaire

- Contexte
- Sommaire des checks
- Les données
 - Les données communes pour tous les checks
 - Les données spécifiques
 - Les données DFE (Duplicate Foreach)
- Comment appliquer un modèle d'hôte à un hôte
 - Application du modèle via l'interface de Configuration
 - Application du modèle via un collecteur d'import de fichiers au format .cfg

Contexte

Le modèle **proxmox-ve-by-API__lifecycle** offre une **vue complémentaire** aux autres modèles du pack, avec des indicateurs spécifiques sur l'état applicatif du nœud Proxmox. L'ensemble des checks apportés par ce modèle d'hôtes permettent d'attester facilement de la bonne configuration du nœud Proxmox.

Ce modèle est efficace pour déterminer rapidement si le nœud Proxmox est correctement mis en place.

Si le besoin de modifier certains éléments (*commandes, checks ou modèles d'hôtes*) se présente, veuillez vous référer à la page ([Les bonnes pratiques d'utilisation d'un pack livré par Shinken](#)).

- Il est possible également de directement modifier le modèle **proxmox-ve-by-API__lifecycle**, car il hérite du modèle **proxmox-ve-by-API__lifecycle__shinken** qui contient toute la logique du modèle.

Vous ne devez **pas modifier les modèles internes** finissant par la particule '**__shinken**', cela risque d'écraser vos modifications lors des mises à jour du pack.

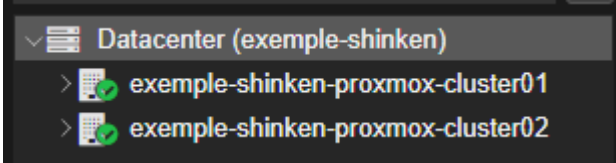
Sommaire des checks

Nom	Description
Node Certificate Expiration by API	Vérifie l'état actuel des certificats utilisés par le nœud Proxmox. (voir la page Node Certificate Expiration by API)
Node License by API	Vérifie la validité de la licence utilisée sur le nœud Proxmox. (voir la page Node License by API)

Les données

Les données communes pour tous les checks

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation du pack	Description
-----	----------------	-------	--------	--	-------------

PROXMOX-VE-BY-API__NODE-NAME	l'Hôte (Onglet Données)	--	\$HOSTNAME\$	\$HOSTNAMES	Le nom du nœud Proxmox supervisé. Il faut utiliser le nom du nœud correspondant à l'adresse IP de l'hôte : - exemple-shinken-proxmox-cluster01 : 172.17.0.14 - exemple-shinken-proxmox-cluster01 : 172.17.0.15  Si l'adresse IP de l'hôte Shinken est 172.17.0.14, alors PROXMOX-VE-BY-API__NODE-NAME doit être exemple-shinken-proxmox-cluster01.
PROXMOX-VE-BY-API__PORT	l'Hôte (Onglet Données)	--	8006	8006	Le port utilisé pour accéder à l'API. Par défaut, il s'agit du même port que celui de l'interface de configuration et de gestion Proxmox.
PROXMOX-VE-BY-API__SSL__USE-SSL	l'Hôte (Onglet Données)	--	True	True	Le mode de gestion SSL : - True : vérification activée avec les autorités de certification connues du système - False : vérification des certificats désactivée /chemin/vers/cert : vérification du certificat pour une autorité de certification spécifique
PROXMOX-VE-BY-API__TOKEN	l'Hôte (Onglet Données)	--	shinken:secret	shinken:secret	Combinaison du nom de la clé API et de la clé API. TOKEN-API-NAME:API-KEY - Exemple : SUP-SHINKEN:d2abf83e-da98-4583-bf4b-f3e7e292dadd
PROXMOX-VE-BY-API__USER	l'Hôte (Onglet Données)	--	shinken@pve	shinken@pve	Nom de l'utilisateur Proxmox à utiliser pour la connexion.

Les données spécifiques

Pas de données spécifiques pour ce modèle

Les données DFE (Duplicate Foreach)

Pas de données DFE pour ce modèle

Comment appliquer un modèle d'hôte à un hôte

Dans les 2 méthodes suivantes, remplacer **mon_modèle** par le modèle voulu :

- proxmox-ve-by-API__lifecycle

Application du modèle via l'interface de Configuration

Dans l'interface de Configuration :

- créer ou éditer un hôte (voir la page [Éditer un Hôte](#)),
- ajouter le modèle "**mon-modèle**" (selon vos besoins) dans la propriété "**Modèles d'hôte hérités**" à l'aide du menu déroulant.

Modèles d'hôte hérités



-- Valeurs sélectionnées --



mon-modèle [2 checks] x

Application du modèle via un collecteur d'import de fichiers au format .cfg

Dans votre fichier de définition de vos éléments à importer via votre collecteur :

- créer ou éditer la définition de votre hôte,
- ajouter la valeur **mon-modèle** (*selon vos besoins*), dans la propriété "**use**",
- importer le contenu du fichier via un collecteur de type "cfg-file-import" (voir la page [Collecteur de type \(cfg-file-import \) - Import depuis des fichiers au format .cfg](#)).

```
define host {  
    host_name    mon_hôte  
    use          mon-modèle  
}
```