

Personnaliser son Pack iis

Sommaire

- Contexte
 - Comment créer de nouveaux checks et commandes ?
 - Les différents arguments du script
 - Les métriques supervisées par défaut
- Exemple
 - Création d'une nouvelle commande
 - Création d'un nouveau modèle d'hôte
 - Création d'un nouveau check appliqué aux modèle d'hôtes

Contexte

Les checks que nous fournissons ne renvoie que des métriques car nous ne pouvons pas faire de corrélation entre ces métriques et le bon fonctionnement du serveur iis.

Si vous voulez ajouter un seuil à ces métriques pour que le check passe en Warning ou en Critical, cette page est là pour vous montrer comment créer de nouveaux checks et de nouvelles commandes. Ainsi qu'une présentation des différents arguments que vous pouvez ajouter aux commandes pour répondre au mieux à vos besoins.

Pour des besoins plus précis, il vous sera nécessaire de créer de nouveaux checks (voir la page [Les bonnes pratiques d'utilisation d'un pack livré par Shinken](#)).

Comment créer de nouveaux checks et commandes ?

En premier lieu nous vous conseillons de créer des clones des commandes et checks existants plutôt que de modifier directement ceux fournis par Shinken, cela vous évitera d'avoir des "Différences" dans Staging.

Vous pouvez cloner les commandes, les checks et le modèle d'hôte apportés par le pack IIS et par la suite modifier les clones créés.

Les pages de documentation suivantes expliquent comment vous devez procéder :

- [Actions de masse \(Administrateur de SI \)](#)
- [Editer un check appliqué à un hôte](#)
- [Les commandes](#)
- [Editer un Hôte](#)

Les différents arguments du script

La sonde "check_wmi_plus" permet de superviser une machine Windows et ses différents service, il possède plus de 80 vérifications, nous avons pour vous isoler les 5 vérifications permettant de superviser un serveur web IIS dans des checks. Nous allons maintenant vous présenter les différents arguments fournis pour personnaliser ces checks et comment les utiliser.

Les arguments suivants ne sont pas configurables dans les commandes de base fournies par Shinken :

- -a : Cet argument permet de spécifier le nom de l'instance du serveur web que vous voulez superviser. Utilisez _Total pour les statistiques totales de tous les sites web, sinon pour monitorer un serveur web en particulier, entrez son nom. (*Par défaut Shinken utilise _Total*)
- --nokeepstate : Cet argument permet **de ne pas conserver** l'état des résultats fournis par le check dans le dossier temporaire du système. Par défaut, sans cet argument, le check va créer un fichier temporaire pour conserver l'état des résultats et faire une moyenne lors des prochains appels, cela permet d'avoir une moyenne plus précise. Mais nous ne vous recommandons pas de conserver l'état lorsque vous utilisez plusieurs Pollers. Car les Pollers ne partageant pas leurs fichiers temporaires et cela peut produire des erreurs ou des résultat incohérents. (*Par défaut Shinken utilise --nokeepstate pour éviter les erreurs lors de l'utilisation de plusieurs Pollers*)

Les arguments suivants sont configurables par des données dans les commandes de base fournies par Shinken :

- -t : Cet argument spécifie le nombre de secondes au-delà duquel la commande est interrompue. Certaines requêtes WMI prennent plus de temps que d'autres et un réseau avec une latence élevée peuvent nécessiter une augmentation de la valeur par défaut qui est de 15 secondes.
- -y : Cet argument spécifie le nombre de secondes sur lequel le check est effectué. Plus le délai est long, plus le calcul sera précis. Valable uniquement si vous spécifiez également l'argument --nokeepstate, c'est-à-dire si vous n'utilisez pas la fonction de conservation de l'état. Les métriques qui dépendent de cette donnée ont un nom qui termine par "PerSec". La valeur par défaut est de 2 secondes.

 Plus le temps donné à l'argument **-y** sera élevé, plus la valeur renvoyée sera précise. Mais cela a un coût sur les ressources utilisées par le Poller exécutant la commande.

Si le temps d'exécution de la commande dépasse la valeur de l'argument **-t**, la commande va s'interrompre avant d'avoir son résultat. Il est recommandé d'affecter une valeur à l'argument **-t** toujours supérieure à celle de l'argument **-y** d'au moins 6 secondes.

 Si le temps donné à l'argument **-t** dépasse les 60 secondes, le check devra être surchargé pour modifier le "temps maximum d'exécution d'un check" à la valeur définie en paramètre de l'argument **-t**. Voir Temps maximum d'exécution d'un check (voir [Editer un check appliqué à un hôte](#)).

Voici les arguments optionnels que nous n'utilisons pas et qui peuvent vous être utile :

- **-w** : Définit à quelle valeur le check passe en Warning.
- **-c** : Définit à quelle valeur le check passe en Critique.

Attention, par défaut pour les arguments **-w** et **-c**, **une seule métrique est supervisée** pour chaque commande, **et non pas toutes**.

Consultez la section "Les métriques supervisées par défaut" présente plus bas dans cette page, pour connaître la métrique supervisée par défaut pour chaque commande.

Vous pouvez personnaliser la métrique supervisée avec la syntaxe suivante :

 **-w** METRIQUE=VALUE
ou
-c METRIQUE=VALUE

METRIQUE étant le nom de la métrique à superviser et **VALUE** la valeur à laquelle le check passe en Warning/Critique.

Pour chacune des commandes du pack IIS vous avez accès à la liste des métriques que renvoie chaque commande. Cette liste est disponible sur la page de documentation de chaque check dans la rubrique Métriques. Voici les liens vers chacune des pages de documentation correspondantes :

- [Connections - Modèle iis](#)
- [Errors - Modèle iis](#)
- [RequestsActivity - Modèle iis](#)
- [TransfertsActivity - Modèle iis](#)
- [Users - Modèle iis](#)

Les métriques supervisées par défaut

Pour chacune des commandes de supervision d'un serveur web IIS, une seule métrique est supervisée, voici celles supervisé par défaut si vous n'en spécifiez aucune :

Check Name	Métriques supervisé par défaut
Connections	CurrentConnections
Errors	_NotFoundErrorsPersec
RequestsActivity	_GetRequestsPersec
TransfertsActivity	_BytesTotalPersec
Users	_LogonAttemptsPersec

Exemple

Création d'une nouvelle commande

Voici un exemple pour une modification de la commande "Connections", le procédé reste le même pour toutes les autres commandes.

- Sélection de la commande que vous voulez modifier :

☐	-- Tous --	-- Tous --	connection	-- Tous --	-- Tous --	Aucun filtre	Aucun filtre	Aucun filtre	-- Tous --
☒		Active	check_iis_connections	non tagué	Faux	iis	shinken-core	05/10/2022 16:29	cfg-file-shinken

- Clone de la commande :

Action de Masse

Cloner [1]

Exécuter

Exécuter avec un commentaire

(Voir la page [Personnaliser une commande livrée par un pack Shinken](#))

- Contenu de la page d'édition de la commande :

Ici nous allons modifier le contenu de la commande, pour plus d'informations concernant le reste des éléments de cette page vous pouvez lire la doc (voir [Les commandes](#)).

Staging > Commande

Commande > commande_iis_connections_warn_ConnectionAttempts

Général *

Expert

Propriété	Valeur
Nom de la Commande *	commande_iis_connections_warn_ConnectionAttempts
Ligne de Commande *	<code>\$PLUGINDIR\$/check_wmi_plus.pl -H "\$HOSTADDRESS\$" -u "\$_HOSTDOMAINUSER\$" -p "\$_HOSTDOMAINPASSWORD\$" -m checkiis -s connections -a "_ Total" --indir=\$WMI_INI_DIR\$ --nokeepstate -y "\$_HOSTIIS_CONNECTIONS_DELAY\$" -t "\$_HOSTIIS_CONNECTIONS_TIMEOUT\$"</code>
Temps maximum d'exécution d'une commande (secondes)	Par défaut [Même comportement que son check]
Seuil d'alerte de l'utilisation CPU (secondes)	Par défaut [Même comportement que son check]
Activé	☐ Vrai ☐ Faux ☒ Par défaut [Vrai]
Pack	iis

- Ajout d'un argument pour que notre check puisse passer en warning :

On ajoute l'argument -w pour que le check puisse passer en Warning, ainsi qu'une donnée "_HOSTIIS_CONNECTIONS_WARN" pour définir le seuil de Warning.

 Lors de la création d'une nouvelle donnée toujours ajouter `_HOST` devant le nom de votre donnée.

```
$PLUGINDIR$/check_wmi_plus.pl -H "$HOSTADDRESS$" -u "$_HOSTDOMAINUSERS$" -p  
"$_HOSTDOMAINPASSWORDS$" -m checkiis -s connections -a "_Total" --inidir=$WMI_INI_DIR$ --nokeepstate -y  
"$_HOSTIIS_CONNECTIONS_DELAYS$" -t "$_HOSTIIS_CONNECTIONS_TIMEOUTS$"  
-w "$_HOSTIIS_CONNECTIONS_WARN$"
```

- Choix de la métrique à superviser :

Comme expliqué dans "Les différents arguments personnalisables", le Warning ne se fera que sur une seule métrique, ici *CurrentConnections*, le check **Connections** contient aussi une métrique *_ConnectionAttemptsPerSec*.

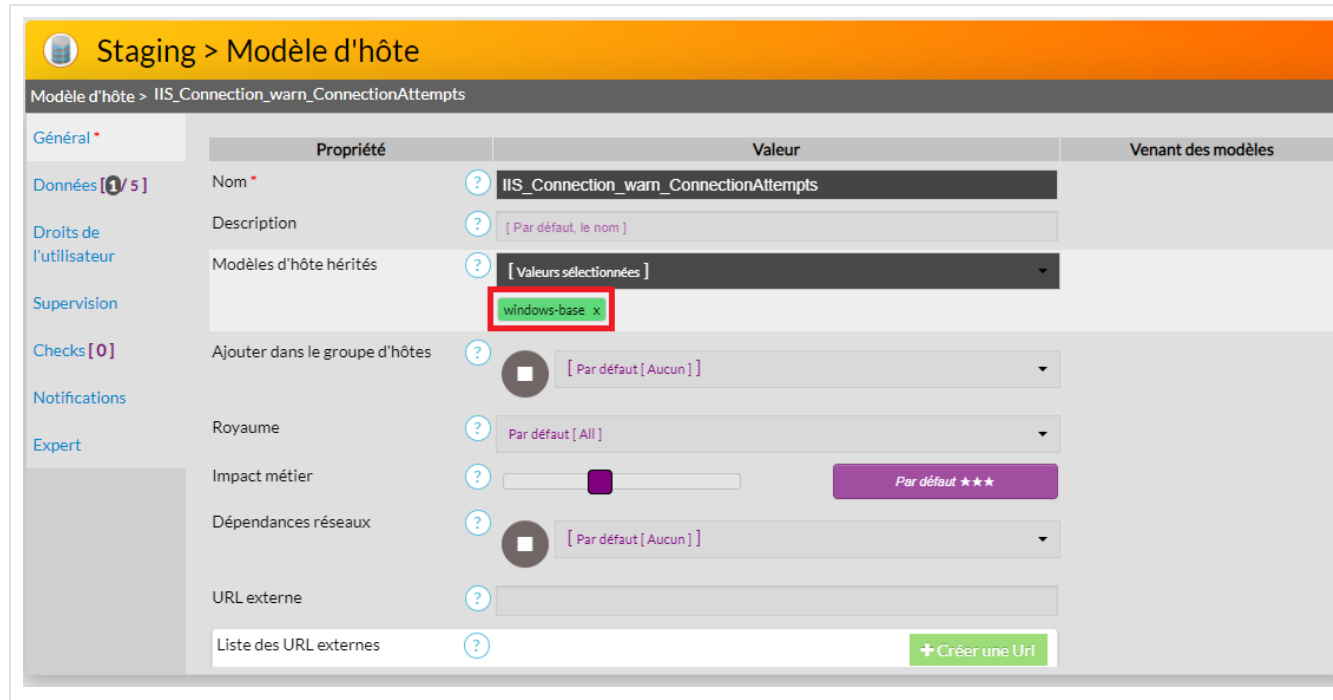
Si vous voulez superviser la métrique *_ConnectionAttemptsPerSec* plutôt que la métrique *CurrentConnections* nous pouvons le spécifier dans notre commande avec "METRIQUE=VALUE", ici *_ConnectionAttemptsPerSec*="\$_HOSTIIS_CONNECTIONS_WARN\$"

```
$PLUGINDIR$/check_wmi_plus.pl -H "$HOSTADDRESS$" -u "$_HOSTDOMAINUSERS$" -p  
"$_HOSTDOMAINPASSWORDS$" -m checkiis -s connections -a "_Total" --inidir=$WMI_INI_DIR$ --nokeepstate -y  
"$_HOSTIIS_CONNECTIONS_DELAYS$" -t "$_HOSTIIS_CONNECTIONS_TIMEOUTS$"  
-w _ConnectionAttemptsPerSec="$_HOSTIIS_CONNECTIONS_WARN$"
```

Création d'un nouveau modèle d'hôte

- Création d'un nouveau modèle d'hôte :

Notre modèle va hériter du modèle Windows-base, le modèle Windows-base précise les données nécessaires pour la connexion à la machine Windows. Vous pouvez aussi faire hériter votre modèle d'hôte du modèle d'hôte IIS pour avoir accès à l'ensemble de ses checks et données.



	Propriété	Valeur	Venant des modèles
Données [1/5]	Nom *	IIS_Connection_warn_ConnectionAttempts	
	Description	[Par défaut, le nom]	
Droits de l'utilisateur	Modèles d'hôte hérités	[Valeurs sélectionnées]	
		windows-base x	
Supervision	Ajouter dans le groupe d'hôtes	[Par défaut [Aucun]]	
Checks [0]	Royaume	Par défaut [All]	
Notifications	Impact métier	Par défaut ★★★	
Expert	Dépendances réseaux	[Par défaut [Aucun]]	
	URL externe		
	Liste des URL externes		+ Créer une Url

- Ajout de la donnée utilisé par notre commande, dans notre modèle :

La donnée que nous utilisons dans notre commande est `_HOSTIIS_CONNECTIONS_WARN`, pour le nom de la donnée de notre modèle d'hôte il faut utiliser le même nom sans le `HOST`. Par défaut la commande `iis_connections` utilise deux autres données, `IIS_CONNECTIONS_DELAY` et `IIS_CONNECTIONS_TIMEOUT`. Nous n'avons pas retiré leur utilisation dans la commande, il faut donc les ajouter à notre modèle d'hôte. Sauf si vous héritez du modèle d'hôte `iis`.

Staging > Modèle d'hôte

Modèle d'hôte > IIS_Connection_warn_ConnectionAttempts

Général

Données [3 / 7]

Droits de l'utilisateur

Supervision

Checks [1]

Notifications

Expert

Données locales & héritées d'un modèle [+ Ajouter une donnée locale](#)

Locale	Nom	Valeur	Venant des modèles
Locale [3 / 3]	IIS_CONNECTIONS_DELAY	5	
Venant des modèles	IIS_CONNECTIONS_TIMEOUT	15	windows-base [4]
	IIS_CONNECTIONS_WARN	100	

(Voir la page [Créer un modèle d'hôte hérité d'un modèle livré dans un pack Shinken](#))

Création d'un nouveau check appliqué aux modèle d'hôtes

- Création d'un nouveau check :

Pour cela nous allons copier notre check `Connections` :

Staging > Les Check appliqué aux r

Action de Masse: Cloner [1] [Exécuter](#) [Exécuter avec un commentaire](#)

Type	Actif	Nom	Attaché sur les modèles d'hôte	Attaché sur les groupes d'hôtes
☒	Activé	Connections	iis	

- Contenu de la page d'édition du check :

1 : Si vous avez apporté des modifications au modèle de Check `iis-service` et que vous souhaitez les conserver sur votre check vous pouvez hériter du modèle de Check `iis-service`, sinon créer votre propre modèle de check ou simplement utiliser le modèle `generic-service`.

2 : Ici vous entrez le nom de votre modèle d'hôte, dans cet exemple c'est `IIS_Connection_warn_ConnectionAttempts`.

- Modifiez le check pour utiliser la commande que nous avons créée :

- Votre modèle d'hôte a maintenant accès à votre nouveau check et sa nouvelle commande.
- Vous pouvez appliquer ce modèle à n'importe quel hôte, pour qu'il ait accès à ce nouveau check.

(Voir la page [Créer un check attaché à un modèle d'hôte hérité d'un modèle livré dans un pack Shinken](#))