

# Modèle Switch-SNMPv1v2-detailed

## Sommaire

- Contexte
- Sommaire des checks
- Les données
  - Les données communes
  - Les données spécifiques
    - Pour le check "Cpu Switch SNMPv1v2"
    - Pour le check "Memory Switch SNMPv1v2"
  - Les données DFE ( Duplicate Foreach )
    - Utilisation
- 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 **Switch-SNMPv1v2-detailed** permet d'avoir un point de vue plus détaillé sur **chaque interface** du switch, et donner des informations plus précises comme la mémoire RAM disponible et l'état du CPU.

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

## Sommaire des checks

Nom	Description
Hardware Health Switch SNMPv1v2	Il vérifie le bon fonctionnement physique du matériel de l'appareil ( <i>alimentation, ventilateurs, températures, disques...</i> ).  ( voir la page <a href="#">Hardware Health Switch SNMPv1v2 - Switch-SNMPv1v2-detailed</a> )
Cpu Switch SNMPv1v2	Il récupère et affiche les informations concernant l'utilisation du/des processeur(s) du switch.  ( voir la page <a href="#">Cpu Switch SNMPv1v2</a> )
Memory Switch SNMPv1v2	Il récupère et affiche les informations sur l'utilisation mémoire du switch.  ( voir la page <a href="#">Memory Switch SNMPv1v2</a> )
<b>\$KEY\$</b> : Errors Switch SNMPv1v2  ( exemple : \$KEY\$ = Vlan201, Vlan202 => Vlan201: Errors Switch SNMPv1v2 => Vlan202: Errors Switch SNMPv1v2 )	Il récupère et affiche le taux moyen d'erreurs en entrée et en sortie des interfaces.  ( voir la page <a href="#">\$KEY\$ Errors Switch SNMPv1v2</a> )  Utilise le mécanisme de DuplicateForEach ( Données SWITCH_INTERFACES )

<p><b>\$KEY\$</b> : Status Switch SNMPv1v2</p> <p>( <i>exemple</i> : <b>\$KEY\$ = Vlan201, Vlan202</b>  =&gt; <i>Vlan201: Status Switch SNMPv1v2</i>  =&gt; <i>Vlan202: Status Switch SNMPv1v2</i> )</p>	<p>Il récupère et affiche les informations concernant le statut des interfaces réseaux de votre switch.</p> <p>( voir la page <a href="#">\$KEY\$ Status Switch SNMPv1v2</a> )</p> <p>Utilise le mécanisme de DuplicateForEach ( <i>Données SWITCH_INTERFACES</i> )</p>
<p><b>\$KEY\$</b> : Usage Switch SNMPv1v2</p> <p>( <i>exemple</i> : <b>\$KEY\$ = Vlan201, Vlan202</b>  =&gt; <i>Vlan201: Usage Switch SNMPv1v2</i>  =&gt; <i>Vlan202: Usage Switch SNMPv1v2</i> )</p>	<p>Il récupère et affiche les informations sur le volume d'utilisation de toutes les interfaces réseaux de votre switch.</p> <p>( voir la page <a href="#">\$KEY\$ Usage Switch SNMPv1v2</a> )</p> <p>Utilise le mécanisme de DuplicateForEach ( <i>Données SWITCH_INTERFACES</i> )</p>

## Les données

### Les données communes

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation de Shinken	Description
SNMPCOMMUNITY	l'Hôte ( Onglet Données )	--	\$SNMPCOMMUNITYREAD\$	<b>public</b>	La Communauté SNMP v1/v2 défini sur le switch supervisé : <ul style="list-style-type: none"> <li>En SNMP v1/v2, la communauté est un équivalent à un ID ou à un mot de passe pour se connecter aux équipements.</li> </ul>
SWITCH_TIMEOUT	l'Hôte ( Onglet Données )	secondes	60	<b>60</b>	Donnée permettant au check de s'arrêter après un certain temps si une tâche ne s'est pas terminée. <ul style="list-style-type: none"> <li>Permet d'éviter que le programme ne s'exécute indéfiniment et de prévenir des problèmes de performances.</li> </ul> <p>( La valeur doit être supérieure à 3 )</p>
SWITCH_PORT	l'Hôte ( Onglet Données )	--	161	<b>161</b>	Donnée correspondant au port SNMP sur lequel le démon SNMP de l'équipement supervisé écoute.
SWITCH_WORKING_FOLDER	l'Hôte ( Onglet Données )		/var/tmp /check_nwc_health	<b>/var/tmp</b> <b>/check_nwc_health</b>	Dossier dans lequel la sonde stockera ses fichiers de travail
SWITCH_CUSTOM_OPTIONS	l'Hôte ( Onglet Données )		-	-	Permet d'utiliser les options supplémentaires de la sonde. <ul style="list-style-type: none"> <li>Exemple : --units MB ( <i>change l'unité des métriques retournées par la sonde</i> )</li> <li>Pour connaître les options disponibles de la sonde exécuter la commande : <pre style="border: 1px solid black; padding: 5px; margin-top: 10px;">/var/lib/shinken/libexec /check_nwc_health --help</pre> </li> </ul>



### Remarque

Par défaut, la donnée SNMPCOMMUNITY prend le contenu de la donnée globale \$SNMPCOMMUNITYREAD\$ dont la valeur est modifiable dans le fichier `/etc/shinken/resource.d/snmp.cfg`,

- voir la page [Les Variables \( Remplacement dynamique de contenu - Anciennement les Macros \)](#), pour comprendre leur utilisation.

La modification des valeurs par défaut présentes dans le fichier du serveur ( `/etc/shinken/resource.d/snmp.cfg` ) nécessite un redémarrage des démons suivants :

- Synchronizer pour le voir dans l'interface,
- Arbiter pour que la modification soit prise en compte en supervision.

Donc, pour changer la valeur de SNMPCOMMUNITY, il faut soit :

- changer sa valeur dans un hôte,
- changer sa valeur dans le modèle Switch-SNMPv1v2 ( *mais cela impactera tous les hôtes utilisant ce modèle* ).
- modifier le contenu de la donnée globale \$SNMPCOMMUNITYREAD\$ ( *en prenant en compte que **tous les hôtes** utilisant cette donnée seront impactés* ).

## Les données spécifiques

### Pour le check "Cpu Switch SNMPv1v2"

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation de Shinken	Description
SWITCH_CPU_LOAD_CRIT	l'Hôte ( Onglet Données )	Pourcentage	90	<b>90</b>	Valeur de charge à partir de laquelle le check passe en CRITIQUE.
SWITCH_CPU_LOAD_WARN	l'Hôte ( Onglet Données )	Pourcentage	80	<b>80</b>	Valeur de charge à partir de laquelle le check passe en AVERTISSEMENT.

### Pour le check "Memory Switch SNMPv1v2"

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation de Shinken	Description
SWITCH_MEMORY_USAGE_CRIT	l'Hôte ( Onglet Données )	Pourcentage	90	<b>90</b>	Valeur de charge à partir de laquelle le check passe en CRITIQUE.
SWITCH_MEMORY_USAGE_WARN	l'Hôte ( Onglet Données )	Pourcentage	80	<b>80</b>	Valeur de charge à partir de laquelle le check passe en AVERTISSEMENT.

## Les données DFE ( Duplicate Foreach )

Nom	Modifiable sur	Unités	Défaut	Valeur par défaut à l'installation de Shinken	Description
-----	----------------	--------	--------	---	-------------

SWITCH_INTERFACES	l'Hôte ( Onglet Données )	---	port1, port2	port1,port2	Nom de ou des interfaces ( <i>séparés par une virgule</i> ), exemple "Vlan01,Vlan02,Vlan03".  Check(s) impacté(s) : <ul style="list-style-type: none"> <li>• \$KEY\$ Errors Switch SNMPv1v2.</li> <li>• \$KEY\$ Status Switch SNMPv1v2.</li> <li>• \$KEY\$ Usage Switch SNMPv1v2.</li> </ul>
-------------------	------------------------------	-----	-----------------	-------------	---

## Utilisation

Il vous faudra saisir les noms des interfaces à surveiller :

- Dans l'exemple ci-dessous, le modèle liste **par défaut** les interfaces appelées **port1** et **port2**.

The screenshot shows a configuration page for 'Locale [0]'. Under 'Venant des modèles', 'Switch-SNMPv1v2-detailed [8]' is selected. Under '(Duplicate Foreach)', 'Switch-SNMPv1v2-detailed [1]' is used by 3 checks. The main configuration table shows:

Nom	Valeur
SWITCH_INTERFACES	port1,port2 [ Dans le modèle Switch-SNMPv1v2-detailed ]

- Changer cette liste avec les noms qui concernent votre équipement :
  - Par exemple Eth1, Eth2, Eth3, pour avoir les checks surveillant ces interfaces.

The screenshot shows the same configuration page as above, but the value for SWITCH\_INTERFACES has been updated to 'Eth1,Eth2,Eth3'.

Nom	Valeur
SWITCH_INTERFACES	Eth1,Eth2,Eth3

Pour cette donnée,

- La **virgule** sert de séparateur.
- Vous pouvez mettre n'importe quelle chaîne de caractères.
  - Si vous voulez surveiller les interfaces réseaux non continues, comme les Eth1, Eth2, Eth3 et Eth9, Eth10, il vous suffit d'en faire la liste :

Eth1,Eth2,Eth3,Eth9,Eth10



### Astuce

Si vous voulez générer une liste de 256 ports de la forme Eth0,Eth1, Eth2, ... Eth255, le faire à la main serait très fastidieux!

Nous avons donné la possibilité de générer **AUTOMATIQUEMENT** les nombres :

- La syntaxe **[nombre1-nombre2]** permet de générer plusieurs valeurs
- donc pour avoir notre liste, saisissez **Eth[0-255]**

Syntaxe	Résultats	Commentaire
<b>Eth[5-9]</b>	Eth5,Eth6,Eth7,Eth8,Eth9	Pour une liste d'interfaces continues
<b>Eth[5-9], Eth[60-65]</b>	Eth5,Eth6,Eth7,Eth8,Eth9,Eth60,Eth61,Eth62,Eth63,Eth64,Eth65	Pour faire des sauts entre plusieurs listes d'interfaces continues
<b>Eth[2-3][0-5]</b>	Eth20,Eth21,Eth22,Eth23,Eth24,Eth25,Eth30,Eth31,Eth32,Eth33,Eth34,Eth35	Pour faire un saut régulier entre des listes d'interfaces

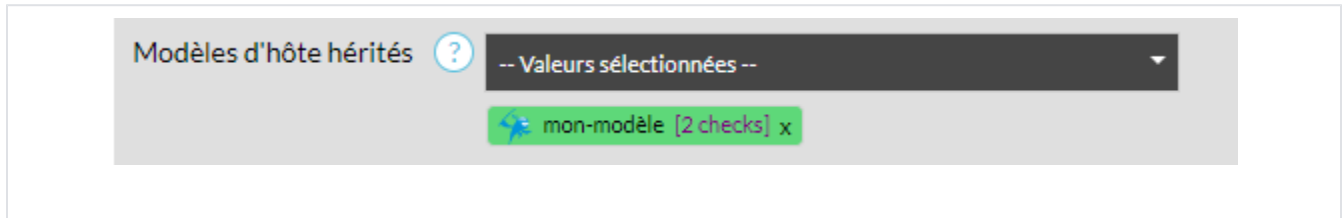
La génération de la liste des valeurs souhaitées se fera par l'Arbiter après la mise en production, elle sera visible uniquement dans l'interface de visualisation, du coté de l'interface de production la syntaxe du DFE ne sera pas interprétée.

## Comment appliquer un modèle d'hôte à un hôte

### 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.



### 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
}
```