

# shinken-scheduler ( Modèle d'hôte )

## Introduction

Ce collecteur vous permet de détecter automatiquement des équipements réseau et des serveurs physiques dans votre infrastructure pour faciliter et accélérer leur import dans la configuration.

Cette source utilise la commande nmap pour la découverte des équipements, pour cela la commande :

- Scanne les machines présentes sur le réseau et détecte les ports ouverts
- Essaye de déterminer le constructeur de l'équipement en fonction de son adresse MAC
- Si possible, détermine son FQDN (Fully Qualified Domain Name).

La source Discovery permet de définir des règles qui, suivant les valeurs remontées par la commande nmap, apportent un complément d'information sur les équipements découverts. Ce complément d'information peut être :

- Des modèles d'hôtes suivant le type d'équipement.
- L'ajout d'un préfixe au nom de l'équipement.

Une fois la découverte exécutée, les équipements détectés et qualifiés sont alors présentés en tant que nouveautés ou différences dans l'interface de Configuration.

Lorsque vous activez le collecteur, il sera non configuré.

- Vous devez entrer dans les pages de configuration. Pour cela, dans la [Page Principale](#), cliquez sur le nom de la source "discovery" pour accéder aux détails de la source.

? Unknown Attachment

- 5 onglets sont disponibles
  - Configuration
  - Règles de découvertes
  - Liste des plages réseaux définies
  - Résumé des dernières exécutions
  - Détail du derniers lancement.

? Unknown Attachment

## Les premiers pas: Réalisons un scan étape par étape

### Etape 1: Editer et ajouter une plage réseau

Les plages réseau scannées par le collecteur discovery peuvent être créées et modifiées dans l'onglet "**Liste des plages réseau définies**",

? Unknown Attachment

#### Sur cette page

[Contexte](#)  
[Description du modèle](#)  
[Checks](#)  
[Données du modèle](#)  
[Métriques enregistrées](#)  
[Commandes](#)  
[Interprétation du check Scheduler - \\$KEY\\$ - Running Well](#)  
[Statistiques de l'état de santé du Scheduler](#)  
[Description des erreurs de Scheduler - \\$KEY\\$ - Running Well](#)  
[Interprétation du check Scheduler - \\$KEY\\$ - Performance](#)  
[Statistiques générales](#)  
[Vol du CPU](#)  
[Suivi des chargements /sauvegardes des données de rétention](#)  
[Suivie des Pollers Satellites](#)  
[Suivie des Reactionners Satellites](#)  
[Type de checks fait par seconde](#)  
[Consommation de temps CPU des checks](#)  
[Cas Particuliers d'erreur](#)  
[Erreur d'un démon bloqué, qui doit être redémarré](#)  
[Le démon a bloqué une tentative de chargement d'objet malveillant](#)  
[Le démon est en cours d'arrêt](#)

le bouton "**+ Ajouter**" permet d'ajouter une nouvelle plage réseau à scanner.

Après avoir cliqué sur le bouton, le formulaire de configuration d'une nouvelle plage réseau va apparaître dans un popup.

Pour créer une plage réseau, vous devez définir les propriétés suivantes:

- **Nom**
- **Plage IP:** Plage(s) d'adresses à scanner dans le format accepté par la commande nmap.

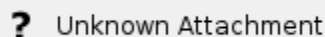
#### Exemples

- 172.16.1.1-254
- 172.16.0.0/24
- 172.16.0.0/24 192.168.1.10-100

- **Plage de ports:** Plage de ports scannés pour chaque adresse. Les 1000 ports les plus répandus sont utilisés par défaut
  - Vous pouvez restreindre le nombre de ports scannés avec une liste.
  - Cette liste peut comporter plusieurs plages en les séparant par des virgules.
  - Ex: **1-1024,2000-8000**
- **Notes:** Texte descriptif au sujet de cette plage réseau
- **Activé:** Activer ou désactiver les scans sur cette plage réseau pour les prochaines exécutions de la source.

## Etape 2: Lancer un scan

Une fois la ou les plages réseau définies, vous pourrez réaliser un scan en utilisant le bouton en haute à droite ( le bouton play ).



Le collecteur va scanner l'ensemble des plages réseau **actives** dans votre configuration.

## Etape 3: Les équipements trouvés

Vous verrez alors le résultat dans l'onglet *Détail* du dernier lancement.

## Le résultat d'un scan (onglet *Détail* du dernier lancement)

Dans l'onglet "*Détail de dernier lancement*" est listé chaque équipement détecté par le collecteur discovery en fonction des plages réseau actives lors de l'import.



Pour chaque équipement, l'œil à droite vous permet de voir le détail de l'opération.

Deux tableaux fournissent respectivement:

- **Les informations collectées par nmap**
  - Toutes les informations présentes dans ce tableau peuvent être utilisées dans les conditions d'une règle.
- **L'Hôte proposé** au Synchronizer:
  - Le collecteur va utiliser certaines données collectées pour les mettre au

format du synchronizer ( Clé / Valeur ).

- Il peut suivant son paramétrage ( des règles par défaut ou définies par l'utilisateur) modifier les valeurs. Cela sera alors mentionné dans la colonne "*Informations supplémentaires*" la règle utilisée.

## Les données collectés par nmap

Les scans réalisés par nmap remontent les clés suivantes:

Clé	Description	Exemple
fqdn	Nom de domaine complètement qualifié	DiskStation
mac	Adresse MAC de l'équipement	00:11:32:9F:09:44
macv endor	Nom du constructeur associé à l'adresse MAC (voir le chapitre suivant pour plus de détails sur la correspondance adresse MAC Constructeur)	Synology Incorporated
open ports	Liste des ports identifiés comme ouverts	22,80,137,139,161,16 1,443,445,548,3261,5 000,5001,5353
os	Famille du système d'exploitation détectée, comme par exemple <i>Windows</i> , <i>Linux</i> , <i>IOS</i> (routeurs Cisco), <i>Solaris</i> ou <i>OpenBSD</i> . Il y a des centaines d'autres familles de systèmes comme des routeurs, imprimantes ou autres systèmes propriétaires.  Lorsque la famille du système d'exploitation ne peut pas être déterminée avec une confiance suffisante, la valeur <i>embedded</i> est utilisée.	Linux
ostype	Le type de système d'exploitation est une classification large selon l'usage prévu de ce système comme " <i>router</i> ", " <i>printer</i> ", ou " <i>game console</i> ". Les systèmes d'exploitation universels tels que Linux et Windows, qui ont de nombreux cas d'utilisations sont classés en tant que " <i>general purpose</i> ".	general purpose
osve ndor	Entreprise ou entité qui produit le système d'exploitation ou équipement (par exemple <i>Apple</i> , <i>Cisco</i> , <i>Microsoft</i> , <i>Link sys</i> ). Pour les projets communautaires comme Linux ou les différents BSD, la valeur de l'information "os" est répétée ici.	Linux
osver sion	Version de l'os détectée	3.X



Si nmap ne peut remplir une information, le message "*Aucune valeur remontée*" sera affiché dans la colonne valeur pour cette clé

## Correspondance entre l'adresse MAC et le constructeur

Lors du scan d'une plage réseau, le collecteur discovery peut remonter le constructeur du matériel à l'aide de nmap.

Cette détection du constructeur se fait par identification de l'adresse MAC de l'équipement détecté sur le réseau. Pour la correspondance entre adresse MAC et constructeur, nmap utilise un fichier nommé *nmap-mac-prefixes* qui comporte des adresses MAC associées à des constructeurs (macvender).

Par exemple, si votre machine récupérée par la discovery a pour adresse MAC "*0050BAXXXXX*", le constructeur détecté (macvender) est "*D-Link*".

Shinken fournit par défaut un fichier *nmap-mac-prefixes* qui sert de référence à nmap. Ce fichier est mis à jour à chaque mise à jour de Shinken.

Pour créer des associations entre adresses MAC et constructeur personnalisées, il est possible créer un fichier *nmap-mac-prefixes* dans **/etc/shinken-user/configuration/daemons/synchronizers/sources/discovery/nmap/**, qui surchargera celui que Shinken met à disposition lors de l'installation.

? Unknown Attachment

Ce fichier doit être au format de l'exemple donné et peut contenir des commentaires en commençant la ligne par un #.

Votre fichier surcharge la liste présente par défaut dans l'installation de Shinken Entreprise.

Le fichier par défaut à utiliser comme modèle est le suivant: [nmap-mac-prefixes](#).

Pour plus d'informations sur la syntaxe à respecter pour ce fichier, la documentation de nmap décrit la syntaxe requise pour ce fichier de préfixes: <https://nmap.org/book/nmap-mac-prefixes.html>

L'exemple suivant fournit une illustration sur la découverte d'un NAS Synology et la détection automatique du constructeur.

? Unknown Attachment

## Les données accrochées à l'hôte proposé au Synchronizer ( Element importé )

Le collecteur accroche les données suivantes à l'hôte proposé au Synchronizer:

Nom		Exemple
_MAC_ADDRESS	Adresse MAC de l'équipement	08:00:27:7A:E6:4A
_SYNC_KEYS	Les clés de synchronisation ( <a href="#">voir la section sur les clés de synchronisation</a> )	<ul style="list-style-type: none"><li>• VM7</li><li>• 172.16.0.7</li></ul>
address	L'address IP de l'équipement	172.16.0.7
host_name	Le nom de l'équipement	VM7
import_date	La date de l'import de l'équipement	16/05/2019 11:11
imported_from	Cette propriété est actuellement utilisée en interne. Dans une future version, ce champ contiendra les détails de la plage qui a été utilisée pour découvrir l'équipement	discovery
source	La source depuis laquelle l'équipement a été importé	discovery
use	Les modèles d'hôtes que la discovery accroche sur l'équipement	http,https,linux,ssh

## Configuration

### Onglet des règles de découverte

Un onglet listant les règles de découvertes (par défaut et définies par l'utilisateur) est disponible dans la page du collecteur discovery.

- Les règles vous permettent de définir des conditions à remplir pour que la discovery accroche automatiquement des modèles d'hôtes sur les équipements remontés.
- Les conditions vont tester les valeurs remontées par nmap.

Les règles sont affichées sous forme de la liste:

- Triée par ordre de priorité: Exemple: Lorsque les règles 1 et 2 s'appliquent, la règle 1 s'applique avant la règle 2 (ajout du préfixe et des modèles d'hôtes)
- La couleur de fond de chaque ligne indique le type de règle:
  - **Blanc**: règle par défaut
  - **Bleu**: règle définie par l'utilisateur
  - **Gris**: règle affichée dans la liste, mais sans effet (désactivée volontairement, syntaxe incorrecte ...).

? Unknown Attachment

Il existe 6 statuts pour les règles de découvertes :

Cas	Statut
La règle est en un seul exemplaire dans votre fichier	Définie par l'utilisateur
Le nom de la règle existe déjà dans le fichier par défaut	Surchargée par l'utilisateur
Le nom de la règle est défini plusieurs fois dans le fichier utilisateur	Définie plusieurs fois
La règle comporte une erreur (comme une des clés obligatoires)	Invalide
La règle ne comporte ni de conditions, ni de modèles d'hôtes, ni de préfixe	Désactivé



Vous pouvez rafraîchir la liste des règles directement en appuyant sur le bouton de rafraîchissement en haut à droite



Unknown Attachment

, ou en appuyant sur F5.

## Définir de nouvelles règles de découvertes ou surcharger les existantes

Le mécanisme de règles permet d'enrichir les équipements détectés.

- Par défaut, une installation fournit une liste de règles prédéfinies.
- Vous pouvez définir vos propres règles ou surcharger les règles prédéfinies. Vous devez pour cela éditer le fichier JSON :
  - `/etc/shinken-user/configuration/daemons/synchronizers/sources/discovery/discovery_rules.json`

Une règle de découverte est séparée en 4 parties:

- name**: le nom et l'id de votre règle (*doit être unique*)
- condition[1-9]**: représente une condition qui applique la règle si elle est remplie (*il suffit qu'une seule condition soit bonne pour que la règle soit appliquée*)
- prefix\_name**: ce préfixe est ajouté au nom des éléments découverts par cette règle (*optionnel*)
- use**: Les modèles d'hôtes ajoutés en cas d'application de la règle. Vous pouvez en mettre autant que vous voulez en les séparant d'une virgule.
  - Les modèles d'hôtes sont ajoutés à la suite de ceux déjà présents sur l'hôte (*ajoutés par d'autres règles*)

### Exemple de règle

```
{
  "rules": [
    {
      "name": "myRuleExample",
      "condition1": "os=myOS AND
osversion=^2",
      "condition2":
"osversion=^2.6.0$",
      "condition3":
"macvendor=^myMacVendor",
      "condition4":
"ostype=myType$",
      "condition5":
"openports=1|2",
      "prefix_name":
"myRulePrefix",
      "use": "myTemplate,
myTemplate2"
    }
  ]
}
```

## Ecriture d'une règle de découverte

Le mécanisme de condition utilise les données collectées par nmap pour modifier l'hôte à proposer au Synchronizer. Les clés du retour nmap sont utilisables pour vos conditions (voir [Les données collectées par nmap](#)).

Il existe plusieurs possibilités pour les conditions de vos règles :

### Commence par (=^...)

Si l'expression commence par '^', la condition signifie que le résultat attendu **doit COMMENCER** par l'expression.

```
macvendor=^myMacVendor
```

### Termine par (=...\$)

Si l'expression termine par '\$', la condition signifie que le résultat attendu **doit TERMINER** par l'expression.

```
ostype=myType$
```

### Est égal (=^...\$)

Si l'expression commence par '^' ET termine par '\$', la condition signifie que le résultat attendu **doit être l'expression EXACTE**.

```
osversion=^2.6.0$
```

### Contient (=...)

Si l'expression ne contient aucun des paramètres précédents, la condition signifie que le résultat attendu **doit CONTENIR** l'expression

```
os=myOS
```

### Condition\_1 ET condition\_2 (condition\_1 AND condition\_2)

Si la condition AND est équivalente à la porte logique AND. Cela signifie que **tout** ce qui est dans cette condition doit être respecté pour que la règle soit appliquée.

```
os=myOS AND osversion=^2
```

### Cas spécifique des openports (X|X)

L'écriture d'une condition pour la propriété openports est un cas spécifique.

Sur cette propriété, les conditions de type "contient, commence par ou termine par" ne peuvent pas être utilisées.

```
openports=1|2
```

- Il faut donc rentrer le port exact.
- La présence des caractères '^' et '\$' sera donc considérée comme une erreur.

Pour faire un OU logique, il faut mettre un '|' entre chaque port.

Exemple: 80|8080

### Liste des règles par défaut

Lors de l'installation, Shinken livre un certain nombre de règles par défaut pour la détection des objets via le collecteur discovery.

Ces règles par défaut sont les suivantes:

Règle	Condition	Modèle d'hôte appliqué
aix	os=aix	aix
cisco	os=cisco	cisco
dns	openports=53	dns
ftp	openports=21	ftp
HPAasm	macvendor=hewlett packard AND openports=2301	hp-asm
HPBladeChassis	os=embedded AND ostype=remote management AND osvendor=hp	hp-blade-chassis
HPPrinterState	openports=631 AND openports=9100	printer-hp
HpUx	os=hp-ux	hpux
Http	openports=80	http
Https	openports=443	https
Imap	openports=143	imap
Imaps	openports=993	imaps
Ldap	openports=389	ldap
Ldaps	openports=636	ldaps

linux	os=linux	linux
mongodb	openports=27017	mongodb
mssql	openports=1433	mssql
mysql	openports=3306	mysql
Oracle	openports=1521   1526	oracle
pop3	openports=110	pop3
pop3s	openports=995	pop3s
smtp	openports=25	smtp
smtps	openports=465	smtps
ssh	openports=22	ssh
Shinken-synchronizer	openports=7765   7766	shinken-synchronizer
Shinken-broker	openports=7767   7772	shinken-broker
Shinken-scheduler	openports=7768	shinken-scheduler
Shinken-reactionner	openports=7769	shinken-reactionner
Shinken-arbiter	openports=7770	shinken-arbiter
Shinken-poller	openports=7771	shinken-poller
Shinken-receiver	openports=7773	shinken-receiver
switch	ostype=switch	switch
ESX	isesxhost=1	esx
VMware-VM	isesxvm=1	vmware-vm
Windows	os=windows	windows
Windows 2000	os=windows AND osversion=2000	windows2000
Windows 2003	os=windows AND osversion=2003	windows2003
Windows 2008	os=windows AND osversion=vista	windows2008
Windows 2008r2	os=windows AND osversion=7	windows2008,windows2008r2
Windows 2012	os=windows AND osversion=2012	windows2012
Windows 2016	os=windows AND osversion=2016	windows2016

Le fichier des règles par défaut est le suivant: [discovery\\_rules.json](#)

## Configuration avancée

Le comportement du collecteur discovery peut être configuré de manière plus précise dans le fichier de configuration de la source.

Ce fichier est disponible au chemin suivant (**/etc/shinken/sources/discovery.cfg**) et contient les propriétés suivantes:

Propriété	Valeur par défaut	Description
<b>source_name</b>	discovery	Nom de la source. Doit être unique ( <b>non modifiable pour le moment</b> )
<b>order</b>	10	Ordre dans la consolidation de l'algorithme pour cette source . Voir dans la page <a href="#">Synchronizer</a> page pour plus d'information
<b>import_interval</b>	5	Intervalle en minutes de chargement de la source.
<b>modules</b>	discovery-import	Module utilisé
<b>enabled</b>	0	1 - Active la source 0 - Désactive la source. Elle est visible dans l'interface, mais ne collecte pas de données.
<b>not_stored_properties</b>	< liste de champs >	Ce paramètre permet de définir un ou plusieurs champs que ne seront pas importés dans shinken. Cela peut être utile pour exclure une propriété ou bien utiliser des champs personnalisés utiles pour la gestion de vos fichiers .cfg
<b>data_backend</b>	mongodb	Base de données où les informations de la source vont être stockées
<b>mongodb_url</b>	<a href="#">mongodb://localhost/?safe=false</a>	URL d'accès à MongoDB
<b>mongodb_database</b>	synchronizer	Base Mongo où sont stockées les données de la source
<b>rules_path</b>	/etc/shinken-user/configuration/daemons/synchronizers/sources/discovery/discovery_rules.json	Fichier json comportant vos règles de découvertes ( <a href="#">voir règles de découvertes</a> )
<b>nmap_mac_prefixes_path</b>	/etc/shinken-user/configuration/daemons/synchronizers/sources/discovery/nmap/nmap-mac-prefixes	Fichier comportant vos propres nmap-mac-prefixes ( <a href="#">voir mécanisme de correspondance entre adresse MAC et constructeur</a> )

#### Exemple de définition:

```
define source {
    source_name      discovery
    order            10
    import_interval  5
    module_type      discovery-import
    data_backend     mongodb
    mongodb_uri      mongodb://localhost/?safe=false
    mongodb_database synchronizer
    rules_path       /etc/shinken-user/configuration/daemons/synchronizers/sources/discovery
/discovery_rules.json
    nmap_mac_prefixes_path /etc/shinken-user/configuration/daemons/synchronizers/sources/discovery/nmap
/nmap-mac-prefixes
}
```

## Précisions techniques

### Sécurité: paramètres de la commande nmap

La commande nmap lancée par la source discovery utilise les paramètres suivants:

- **-PE** : Ping Scan (Echo Request)
- **-sU** : Scan UDP
- **-sT** : Scan TCP
- **--min-rate 1000** : Envoie un minimum de 1000 paquets par secondes
- **--max-retries 3** : Effectue au maximum 3 retransmissions en cas d'erreur sur les scan de ports
- **-T4** : Optimisation de performances
- **-O** : Détection des systèmes d'exploitation
- **-oX** : Export XML (utilisé pour l'interprétation des données par Shinken)

## Clés de synchronisation

Les clés de synchronisation sont des propriétés des objets utilisées pour les identifier dans les sources. Le fonctionnement et l'utilité des clés de synchronisation sont décrits de manière plus détaillée dans la page de documentation dédiée: [Précision technique sur le fonctionnement de l'import des sources](#).

Les informations suivantes de la découverte réseau sont ajoutées en tant que clés de synchronisation de l'objet dans Shinken:

- host\_name
- address

## Résolution des problèmes courants

### Base de données inaccessible

Si votre discovery n'arrive pas à accéder à la base de données, elle devient alors indisponible. Pendant cette indisponibilité, il est impossible d'effectuer quelques manipulations :

- Voir la liste des règles
- Voir la liste des plages réseau
- Ajouter une plage réseau

Rafraîchir la page ou lancer un import permet de réessayer d'accéder à la base de données.

? Unknown Attachment

? Unknown Attachment

### Le fichier de règles n'est pas correctement chargé

Il peut y avoir plusieurs problèmes à l'ouverture de votre fichier de règles:

Problèmes	Résultat
Votre fichier n'est pas dans un format json valide	Le fichier n'est pas lu et une erreur apparaît
Votre fichier est introuvable	Le fichier n'est pas lu et une erreur apparaît
Votre fichier est vide	Le fichier n'est pas lu mais aucune erreur n'apparaît

? Unknown Attachment

#### ! Erreurs listées pour la recherche sur cette page

- Le fichier de règles utilisateur () n'est pas dans un format JSON valide
- Le fichier de règles utilisateur () n'a pas été trouvé
- Une erreur est parvenue avec le fichier de règles utilisateur ()

Dans le cas où le fichier de règles n'est pas correctement chargé ([voir comment définir de nouvelles règles de découvertes ou surcharger les existantes](#)), seules les règles par défaut sont prises en compte et un message d'erreur apparaît en haut du tableau.

### Le fichier de préfixes nmap n'est pas chargé

Il peut y avoir plusieurs problèmes à l'ouverture de votre fichier de préfixes nmap qui empêcheront le collecteur discovery de lancer son import, dans ce cas-là, une erreur s'affichera dans la page principale, et vous pourrez avoir plus d'informations en cliquant sur le lien à côté du résultat de votre source, ce qui vous amènera dans l'onglet de **Résumé des dernières exécutions** de votre source:

Problèmes	Résolution
Votre fichier est introuvable	Vérifiez que l'emplacement et le nom de votre fichier correspond à celui renseigné dans le fichier de configuration de votre source discovery.
Votre fichier comporte des erreurs de syntaxe	Vérifiez que la syntaxe du fichier correspond bien à la syntaxe utilisée par nmap décrite dans la documentation suivante: <a href="https://nmap.org/book/nmap-mac-prefixes.html">https://nmap.org/book/nmap-mac-prefixes.html</a>

? Unknown Attachment

? Unknown Attachment

? Unknown Attachment



#### Erreurs listées pour la recherche sur cette page

- Le fichier utilisateur nmap-mac-prefixes n'a pas été trouvé ()
- Le fichier comporte des erreurs de syntaxe, vérifiez son format.