

Le mapping des informations collectées des champs du serveur Open LDAP vers les propriétés et les données Shinken

Sommaire

- [Qu'est-ce qu'une règle de mapping OpenLDAP ?](#)
- [Définition d'une règle de mapping utilisateur](#)
 - [Forcer la valeur des propriétés](#)
 - [Liste des clés d'import pouvant être forcées](#)
 - [Chemin du fichier de définition des règles de mapping](#)
- [Exemple](#)
- [Compatibilité avec les anciennes versions](#)

Qu'est-ce qu'une règle de mapping OpenLDAP ?

Une règle de mapping permet de mettre en relation, lors de l'import, un attribut LDAP avec une propriété ou une donnée Shinken en fonction du type de l'élément (*hôte, groupe d'hôtes, utilisateur, etc.*).

- Une règle peut s'appliquer pour tous les types d'éléments présents dans l'interface de Shinken.

Définition d'une règle de mapping utilisateur

Chaque ligne de ce fichier définit une correspondance entre une propriété ou donnée Shinken et un attribut LDAP.

- Les paramètres peuvent donc être nombreux.
- Par exemple, la ligne suivante définit une correspondance entre le nom de l'hôte dans Shinken et son attribut dans LDAP :

```
"host.host_name" : "name",
```

Elle est définie par :

- La propriété Shinken entre guillemets découpée en deux parties :
 - Le type de l'élément au singulier. Les types disponibles sont : "host", "contact", "hostgroup" et "contactgroup".
 - Le point permet de joindre les deux parties.
 - La propriété ou la donnée de l'élément. Cette propriété ou donnée est la "clé d'import" que l'on peut retrouver sur l'interface dans la fenêtre d'aide.
- Le séparateur "deux points".
- L'attribut LDAP entre guillemets.
- Une virgule, s'il ne s'agit pas de la dernière ligne non commentée.



Si ce format n'est pas respecté, l'import se déroulera sans cette ligne et un message d'avertissement indiquera qu'il y a une erreur de syntaxe.

Il est donc possible de choisir avec quelles propriétés Shinken on peut faire correspondre les attributs LDAP. Si ce fichier n'est pas défini, un mapping par défaut sera utilisé. Voici ses valeurs :

Propriété Shinken	Propriété OpenLDAP	Description
host.host_name	cn	L'attribut LDAP "cn" sera utilisé pour le nom des hôtes.
host.display_name	description	L'attribut LDAP "description" sera utilisé pour la description des hôtes.
host.address	ipHostNumber	L'attribut LDAP "ipHostNumber" sera utilisé pour l'adresse des hôtes.
hostgroup.hostgroup_name	cn	L'attribut LDAP "cn" (CommonName) sera utilisé pour le nom des groupes d'hôtes.
contact.contact_name	uid	L'attribut LDAP "uid" sera utilisé pour le nom des contacts.
contact.email	mail	L'attribut LDAP "mail" sera utilisé comme adresse e-mail du contact.

contact.display_name	displayName	L'attribut LDAP "displayName" sera utilisé comme description du contact.
contact._PHONE	telephoneNumber	L'attribut LDAP "telephoneNumber" sera conservé dans la donnée _PHONE du contact.
contact._MOBILE	mobile	L'attribut LDAP "mobile" sera conservé dans la donnée _MOBILE du contact.
contact._COUNTRY	co	L'attribut LDAP "co" (Country) sera conservé dans la donnée _COUNTRY du contact.
contact._CITY	l	L'attribut LDAP "l" (localityName) sera conservé dans la donnée _CITY du contact.
contact._COMPANY	company	L'attribut LDAP "company" sera conservé dans la donnée _COMPANY du contact.
contactgroup. contactgroup_name	cn	L'attribut LDAP "cn" (CommonName) sera utilisé pour le nom des groupes de contacts.



Pour ne pas importer les propriétés par défaut, un fichier de "mapping" doit être défini avec la clé souhaitée et sa valeur à vide.

Forcer la valeur des propriétés

Il est possible d'ajouter le suffixe **[FORCE]** (*sans rajouter d'espace*) à la **clé d'import** d'un élément Shinken afin de forcer la valeur associée (voir la page [Forcer la valeur des noms des éléments et des propriétés de type liste \(comme la propriété des modèles hérités\)](#)).

Exemple :

```
"host.address[FORCE]": "ipHostNumber",
```

Lorsqu'une propriété est marquée avec **[FORCE]**, elle est traitée de la manière suivante par le mélange des sources :

- La valeur est considérée comme prioritaire par rapport aux autres sources.
- Si une autre source importe le même objet avec une valeur différente, cette dernière est ignorée.
- En cas de conflit entre plusieurs valeurs forcées, la priorité de la source détermine la valeur conservée.

Ajouter **[FORCE]** à une clé d'import contenant une liste (*comme la propriété "members" servant à définir les membres d'un groupe d'utilisateurs*) aura pour effet de remplacer complètement la liste des autres sources par celle fournie avec l'option **[FORCE]**.

Liste des clés d'import pouvant être forcées

- Les clés d'import définissant les noms des éléments
 - Exemple : "host_name" pour les Hôtes.
- Et toutes les propriétés pouvant avoir des valeurs multiples : (voir la page [Éditer les Eléments \(hôte, clusters, checks, utilisateurs ... \)](#))
 - Exemple : "view_contacts" pour la propriété "Les utilisateurs qui voient l'hôte".

La liste exhaustive de toutes les clé est disponible ici : [Liste des clés pouvant utiliser \[FORCE\]](#)

Chemin du fichier de définition des règles de mapping

Pour ajouter une règle utilisateur d'application des modèles, il faut éditer le chemin suivant :

```
/etc/shinken-user/source-data/source-data-[nom de la source]/_configuration/openldap-mapping.json
```

Exemple de chemin pour la source livrée par défaut :

```
/etc/shinken-user/source-data/source-data-openldap-sample/_configuration/openldap-mapping.json
```

Exemple

/etc/shinken-user/source-data/source-data-openldap-sample/_configuration/openldap-mapping.json

```
# IMPORTANT: Do not edit this file.
# To have your own mapping, copy it under the /etc/shinken-users/source-data/YOUR_SOURCE_FOLDER
/_configuration/ directory and edit your copy instead.
# Note: comments should be with a # starting the line, NOT after a value
{
  # You can map any ldap attribut in a Data (start with a _ and in MAJ)
  # -- Hosts
  # mandatory
  "host.host_name": "cn",
  "host.display_name": "description",
  "host.address": "ipHostNumber",
  "host._OS": "operatingSystem",

  # -- Hostgroups
  # mandatory
  "hostgroup.hostgroup_name": "cn",

  # -- Contacts
  # mandatory
  "contact.contact_name": "uid",

  "contact.email": "mail",
  "contact.display_name": "displayName",
  "contact._MEMBER": "uniqueMember",
  "contact._PHONE": "telephoneNumber",
  "contact._MOBILE": "mobile",
  # Co: for country
  "contact._COUNTRY": "co",
  # l: for city
  "contact._CITY": "l",
  "contact._COMPANY": "company",

  # -- Contactgroups
  # mandatory
  "contactgroup.contactgroup_name": "cn"
}
```

Compatibilité avec les anciennes versions

Pour des raisons de compatibilité avec les versions antérieures à la 02.06.03, les propriétés suivantes sont toujours fonctionnelles :

Ancienne propriété	Nouvelle propriété Shinken correspondante	Description
host.name	host.host_name	Contient le nom de l'élément.
host.dNSHostName	host.address	L'adresse de l'élément.
host.operatingSystem	host._OS	Le système d'exploitation sera conservé dans une donnée de l'hôte.
host.operatingSystemServicePack	host._OS_SP	Le service pack du système d'exploitation sera conservé dans une donnée.
host.distinguishedName	host.display_name	Le nom distingué de l'hôte correspondra à la description de l'hôte.
contact.name	contact.contact_name	Contient le nom de l'utilisateur.
contact.mail	contact.email	Contient l'adresse e-mail de l'utilisateur.
contact.displayName	contact.display_name	Contient la description de l'utilisateur.
contact.telephoneNumber	contact._PHONE	Le numéro de téléphone de l'utilisateur sera conservé dans une donnée.

contact.mobile	contact._MOBILE	Le numéro de mobile de l'utilisateur sera conservé dans une donnée.
contact.co	contact._COUNTRY	Le pays de l'utilisateur sera conservé dans une donnée.
contact.l	contact._CITY	La ville de l'utilisateur sera conservée dans une donnée.
contact.company	contact._COMPANY	La société de l'utilisateur sera conservée dans une donnée.
hostgroup.name	hostgroup.hostgroup_name	Contient le nom du groupe.
contactgroup.name	contactgroup. contactgroup_name	Contient le nom du groupe.
contact.member	contactgroup.members	Dans les précédentes versions, les groupes de contacts n'étaient pas importés. Désormais ils le sont et les liens avec les utilisateurs sont conservés.



Les filtres sur les types d'éléments étaient présents dans le fichier de mapping : "host.filter", "contact.filter", "hostgroup.filter" et "contactgroup.filter". Ceux-ci doivent désormais se trouver dans le fichier de connexion sous les noms : "hosts_filter", "contacts_filter", "hostgroups_filter" et "contactsgroups_filter".

- Si les paramètres du fichier de connexion sont définis, ils seront prioritaires.
- Si les paramètres sont absents du fichier de connexion, ceux du fichier de mapping seront pris en compte.



Les paramètres "**contact.classFilter**" et "**contact.categoryFilter**" sont dépréciés et seront supprimés dans une version ultérieure. Le paramètre `contacts_filter` du fichier de connexion doit être utilisé pour remplacer ces filtres.

Exemple : `"contacts_filter" : "(objectClass=InetOrgPerson)"`