Création d'un écouteur

Vue d'ensemble

Ce document décrit comment vous pouvez importer des hôtes et des utilisateurs depuis OpenLDAP.

Après avoir pris soin de lire notre page sur la Définition et Utilisation des sources, certaines étapes sont nécessaires afin de pouvoir importer des hôtes et des utilisateurs depuis un annuaire OpenLDAP :

- Personnaliser la source OpenLDAP (à partir de la source d'exemple)
- Configurer les 3 fichiers pour :
  - la connection à OpenLDAP pour les extractions de données
  - le "mapping" pour la correspondance entre les champs de l'annuaire OpenLDAP et ceux de Shinken
  - les règles de configuration pour appliquer des filtres si besoin et réaliser des actions

Sommaire

Personnaliser la source pré-installée

Le script d'installation et de mise à jour de Shinken permet de mettre en place une source Open LDAP déjà listé (openldap-exemple).

? Unknown Attachment

Vous pouvez la voir sur la page d'accueil de l'UI de Configuration, dans le tableau des sources.

Cette source utilise 2 sortes de fichiers de configuration :

- **Le fichier de définition de source OpenLDAP**
  - Pour OpenLDAP, le fichier en place pour l'exemple est /etc/shinken/sources/openldap.cfg
- **3 fichiers de configuration de la source afin de personnaliser les extractions de données**
  - Disponible dans le répertoire /etc/shinken-user/source-data/source-data-openldap-sample/_configuration :
    - connection_configuration_file,
    - mapping_configuration_file,
    - rules_configuration_file.

Vous pouvez directement utiliser cette source "openldap-example" en modifiant uniquement les fichiers de configuration pour extractions de vos données OpenLDAP (pour un test rapide par exemple). Cependant, nous vous conseillons de dupliquer la source "openldap-exemple" en suivant l'explication ci-dessous.

Pour personnaliser votre source (et donc modifier le terme "example"), **copiez votre répertoire de configuration de la source** (*source-data-openldap-sample* vers *source-data-openldap-monServeurOpenLDAP* par exemple) et modifiez votre définition de source (le nom de votre source et les chemins à vos 3 fichiers de configuration).

Bien entendu, votre Synchronizer devra appeler ce nouveau nom de source, modifiez donc également le fichier de configuration du Shinken Synchronizer.

Définition de la source

Le fichier préalablement créé pour la source OpenLDAP exemple est : /etc/shinken/sources/openldap.cfg. (de la même manière qu'avec le répertoir de configuration, vous pouvez dupliquer ce fichier et effectuer les modifications afin de personnaliser votre source)

C'est ici que vous pouvez changer le nom de votre source via la propriété **source_name**

Une source OpenLDAP est caractérisée par son module_type qui doit être : **ldap-import**

Le module pouvant être utilisé aussi par les sources de type active-directory, il est nécessaire de préciser le mode : **openldap**

Pour les autres valeurs, merci de vous référer à Créer et organiser ses sources.

Les fichiers suivants sont utilisés pour configurer votre serveur OpenLdap (le chemin peut varier si vous avez personnalisé votre répertoire) :

Property	Value	Description
connection_configuration_file	/etc/shinken-user/source-data/source-data-openldap-*sample*/_configuration/openldap-connection.json	***Connexion LDAP*** Information de connexion à l'annuaire LDAP. Ce fichier est à modifier obligatoirement
mapping_configuration_file	/etc/shinken-user/source-data/source-data-openldap-*sample*/_configuration/openldap-mapping.json	***Règles de mapping*** Le mapping d'attributs peut être différent d'un annuaire OpenLDAP à l'autre. Par exemple, vous pouvez spécifier dans ce fichier quel serait le nom d'attribut du numéro de téléphone des utilisateurs. Ce fichier est facultatif et peut être utilisé tel quel
rules_configuration_file	/etc/shinken-user/source-data/source-data-openldap-*sample*/_configuration/openldap-rules.json	***Règles de configuration*** Vous pouvez choisir quelles sortes d'hôtes et d'utilisateurs seront récupérés et permet également la définition de critères afin d'appliquer automatiquement des modèles. Ce fichier est facultatif mais indispensable si vous souhaitez personnaliser les éléments remontés.

Configuration de la connexion

? Unknown Attachment	Veuillez donc tout d'abord modifier le fichier **openldap-connection.json :**

Paramètres du fichier

Propriété	Défaut	Obligatoire	Description
url		OUI	Adresse de votre serveur OpenLDAP
ldap_protocol	3	OUI	Version du protocole LDAP (par défaut à 3 si pas spécifié)
base		OUI	OU (Organisation Unit) base pour la découverte de vos objets
username		OUI	Utilisateur utilisé pour la connexion au serveur OpenLDAP
password		OUI	Mot de passe utilisé pour la connexion au serveur OpenLDAP
hosts_base			OU (Organisation Unit) base pour la découverte de vos hôtes. Si ce paramètre est absent ou vide, aucun hôte ne sera découvert.
hosts_filter	(|(objectClass=device)(objectClass=computer))		Filtre au format ldap utilisé pour découvrir uniquement certains hôtes.
hosts_filter_with_group			Permet de ne filtrer que les hôtes présent dans des groupes définis.
contacts_base			OU (Organisation Unit) base pour la découverte de vos contacts. Si ce paramètre est absent ou vide, aucun contact ne sera découvert.
contacts_filter	(|(objectClass=InetOrgPerson)(objectClass=user))		Filtre au format ldap utilisé pour découvrir uniquement certains contacts.
contacts_filter_with_group			Permet de ne filtrer que les contacts présent dans des groupes définis.

hostgroups_base			OU (Organisation Unit) base pour la découverte de vos groupes d'hôtes. Si ce paramètre est absent ou vide, aucun groupes d'hôte ne sera découvert.
hostgroups_filter	(|(objectclass=group)(objectclass=groupofnames)(objectclass=groupofuniquenames))		Filtre au format ldap utilisé pour découvrir uniquement certains groupes d'hôtes.
contactgroups_base			OU (Organisation Unit) base pour la découverte de vos groupes de contacts. Si ce paramètre est absent ou vide, aucun groupes de contacts ne sera découvert.
contactgroups_filter	(|(objectClass=groupOfUniqueNames)(objectClass=groupOfNames)(objectClass=posixGroup))		Filtre au format ldap utilisé pour découvrir uniquement certains groupes de contacts.

Si vous ne souhaitez pas importer d'objets OpenLDAP "computer" et donc de ne pas créer d'hôtes en "nouveau" dans Shinken, vous pouvez ne pas définir la propriété **hosts_base** ou bien la laisser vide.
Si vous ne souhaitez pas importer d'objets OpenLDAP "contact" et donc de ne pas créer d'utilisateurs en "nouveau" dans Shinken, vous pouvez ne pas définir la propriété **contacts_base** ou bien la laisser vide

Exemple

/etc/shinken-user/source-data/source-data-active-directory-sample/_configuration/openldap-connection.json

```
{
  # Mandatory
  "url": "ldap://vm-w2k8r2.shinkendom.local/",
  "ldap_protocol":3,
  "base": "dc=shinkendom,dc=local",
  "username": "administrateur@shinkendom.local",
  "password": "P@ssword1",

  # Optionnal
  "hosts_base": "OU=Serveurs,dc=shinkendom,dc=local",
  "hosts_filter": "(|(objectClass=device)(objectClass=computer))",
  "hosts_filter_with_group":"",

  "contacts_base": "OU=Users,dc=shinkendom,dc=local",
  "contacts_filter": "(|(objectClass=InetOrgPerson)(objectClass=user))",
  "contacts_filter_with_group": "",

  "hostgroups_base": "OU=Serveurs,dc=shinkendom,dc=local",
  "hostgroups_filter": "(|(objectclass=group)(objectclass=groupofnames)(objectclass=groupofuniquenames))",

  "contactgroups_base": "OU=Users,dc=shinkendom,dc=local",
  "contactgroups_filter": "(|(objectClass=groupOfUniqueNames)(objectClass=groupOfNames)
(objectClass=posixGroup))"
 }
```

Tip

Le compte utilisé pour envoyer des requêtes LDAP au serveur n'a besoin que d'un accès en "lecture seule". Vous devriez créer un compte de service OpenLDAP dédié à cet accès Shinken.

Il est possible que l'utilisateur ldap utilisé soit soumis à certaines limites (nombre d'entrées, délai, taille, ...). Si cette limite est rencontrée, aucun objets ne sera importé.

Configuration des règles de "mapping"

Il est possible de faire correspondre certaines propriétés OpenLDAP avec des propriétés ou donnée d'un élément Shinken. Il existe un "mapping" pa
défaut pour OpenLDAP, mais il est possible de personnaliser les correspondance.

? Unknown Attachment

Si besoin (facultatif), modifiez le fichier **openldap-mapping.json**

Paramètres du fichier

Chaque ligne de ce fichier définit une correspondance entre une propriété ou donnée Shinken avec une propriété ldap. Les paramètres peuvent don
être nombreux.

Par exemple la ligne suivante définit une correspondance entre le nom de l'hôte dans shinken et sa propriété dans ldap :

```
"host.host_name": "cn",
```

Elle est définit par :

- la propriété shinken entre guillemets découpé 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 de l'élément. Cette propriété est la "clé d'import" que l'on peut retrouver sur l'interface dans la fenêtre d'aide
- le séparateur "deux points"
- la propriété ldap entre guillemets

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

Il vous est donc possible de faire correspondre les propriétés ldap avec les propriétés shinken de votre choix. Si vous ne définissez pas ce fichier, ur
mapping par défaut sera utilisé. Voici ses valeurs.

Propriété Shinken	Propriété OpenLdap	Description
host.host_name	cn	La propriété ldap "cn" (CommonName) sera utilisé pour le nom des hôtes
host.display_name	description	La propriété ldap "description" sera utilisé pour la description des hôtes
host.address	ipHostNumber	La propriété ldap "ipHostNumber" sera utilisé pour l'adresse des hôtes
hostgroup.hostgroup_name	cn	La propriété ldap "cn" (CommonName) sera utilisé pour le nom des groupes d'hôtes
contact.contact_name	uid	La propriété ldap "uid" (UserId) sera utilisé pour le nom des contacts
contact.email	mail	La propriété ldap "mail" sera utilisé comme adresse mail du contact
contact.display_name	displayName	La propriété ldap "displayName" sera utilisé comme description du contact
contact._PHONE	telephoneNumber	La propriété ldap "telephoneNumber" sera conservé dans la donnée _PHONE du contact
contact._MOBILE	mobile	La propriété ldap "mobile" sera conservé dans la donnée _MOBILE du contact
contact._COUNTRY	co	La propriété ldap "co" (Country) sera conservé dans la donnée _COUNTRY du contact
contact._CITY"	l	La propriété ldap "l" (localityName) sera conservé dans la donnée _CITY du contact
contact._COMPANY	company	La propriété ldap "company" sera conservé dans la donnée _COMPANY du contact
contactgroup. contactgroup_name	cn	La propriété ldap "cn" (CommonName) sera utilisé pour le nom des groupes d'hôtes

Si vous ne souhaitez pas importer les propriétés par défaut, vous devez définir un fichier de "mapping" avec la clé souhaité et sa valeur à vide

Exemple

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

```
{
    # 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",


    # -- 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 inférieur 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 de 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 email 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 conservé dans une donnée
contact.company	contact._COMPANY	La société de l'utilisateur sera conservé 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 précédents versions, les groupes de contact n'étaient pas importés. Désormais ils le sont les et liens avec les utilisateurs sont conservés

Les fitres sur les types d'éléments étaient présent 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 nom : "hosts_filter", "contacts_filter", "hostgroups_filter" et "contactsgroups_filter".

- Si les paramètres du fichier de connexion sont définit, ils seront prioritaire
- Si les paramètres sont absent 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. Veuillez utiliser le paramètre contacts_filter du fichier de connexion pour remplacer ce filtre.

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

Règles de configuration

Ce fichier est utilisé pour appliquer des **modèles d'hôtes** et des **modèles d'utilisateurs** sur les hôtes et utilisateurs durant l'import, provenant d'OU ciblées.
Ce fichier permet également de filtrer les utilisateurs à importer dans Shinken en se basant sur les membres d'un ou de plusieurs groupes d'utilisateurs de l'annuaire OpenLDAP ciblé.

? Unknown Attachment	Veuillez donc enfin modifier le fichier **openldap-rules.json** **Attention la modification de ce fichier est obligatoire car certaines propriétés contenant des chemins OpenLDAP ne permettront pas un import valide si elles ne sont pas modifiées.**

Paramètres du fichier

Les règles doivent être définies dans le fichier "rules", encadré par des accolades "{ ... }". Il est possible de définir plusieurs types de règles.

Ajouter un modèle sur tous les éléments

Il est possible d'ajouter un même modèle sur tous les hôtes ou contacts importés par la source. Par exemple pour ajouter le modèle "ldap-host" à tous les hôtes et le modèle "ldap-user" à tous les contacts, il faut configurer les lignes suivantes :

```
"hosts_template": "ldap-host",
"contacts_template": "ldap-user"
```

Ajouter un modèle sur tous les éléments présent dans une OU spécifique

Il est possible d'ajouter un modèle pour tous les hôtes ou contacts qui sont dans une OU spécifique. Par exemple pour ajouter le modèle d'hôte linux à tous les hôtes présent dans l'OU "ou=Linux,ou=Datacenter,dc=shinkendom,dc=local", et le modèle "Bordeaux" à tous les contacts présent dans l'OU "ou=Users,ou=Bordeaux,ou=Datacenter,dc=shinkendom,dc=local", il faut configurer les lignes suivante :

```
"hosts_template_linux": "ou=Linux,ou=Datacenter,dc=shinkendom,dc=local",
"contacts_template_ldapAdmins": "ou=Users,ou=Bordeaux,ou=Datacenter,dc=shinkendom,dc=local"
```

Ajouter un modèle sur tous les éléments correspondant à une propriété

Il est possible d'ajouter des modèles sur des éléments dont un propriété correspond à une valeur définie.

Cette règle est divisé en 5 parties : un préfixe, un modèle, type d'élément, le nom de la propriété, la valeur de la propriété.

Voici un exemple et son explication :

```
exemple        :               "AddLast_template_(France)_to_contact_matching_[_COUNTRY]": "France"
explication :         | Préfixe         |modèle|    type d'élément    |propriété|    valeur |
```

- Tout d'abord le préfixe peut prendre 3 valeurs :
  - "AddFirst_template_" : permet d'ajouter le modèle en première position du champs use
  - "AddLast_template_" : permet d'ajouter le modèle en dernière position du champs use
  - "Force_template_" : permet d'utiliser uniquement ce modèle. Cette méthode sera prioritaire et effacera les valeurs obtenues par les précédentes règles.
- Le nom du modèle doit être mis entre parenthèses : France dans l'exemple
- Le type d'élément ne peut prendre que deux valeurs
  - "_to_host_matching_" : la règle sera utilisé pour les hôtes
  - "_to_contact_matching_" : la règle sera utilisé pour les contacts
- le nom de la propriété doit être entre crochets : _COUNTRY dans l'exemple
  - Cette propriété peut être une propriété ou donnée de l'élément shinken (présente dans le fichier de mapping)
  - Cette propriété peut être un attribut ldap
- la valeur de la propriété pour que la règle s'applique : France dans l'exemple

Ajouter un modèle sur tous les élément présent dans un groupe

Pour ajouter un modèle sur un élément présent dans un groupe, il faut utiliser la règle précédente avec la propriété memberOf.

Sur certaines installation openLDAP, l'overlay memberOf n'est pas installé et n'est pas utilisable nativement. Le module import-ldap n'utilise pas cet overlay et le simule. Vous pouvez donc utiliser la propriété memberOf dans ce module même s'il n'est pas présent sur votre installation.

Propriétés multiples et règles

Dans LDAP, il est possible qu'un objet possède plusieurs fois la même propriété. Dans ce cas, il est possible de mettre en place des règles sur ces propriétés. Par exemple, un considère un objet qui a plusieurs propriétés "l" (location) (un utilisateur qui travaille à plusieurs endroits par exemple).

On peut alors mettre en place la règle suivante:

openldap-rules.json

```
"AddLast_template_(bordeaux)_to_host_matching_[l]": "Bordeaux"
```

Cette règle est activée lorsque l'utilisateur possède une de ses propriétés "l" qui est égale à "Bordeaux". Dans ce cas, le modèle d'utilisateur "bordeaux" est utilisé pour cet utilisateur.

Exemple

Dans cet exemple nous allons ligne par ligne :

- ajouter le modèle "linux" à tous les hôtes decouvert par la source
- ajouter le modèle "centos" à tous les hôtes découvert dans l'OU "OU=centos,OU=serveurs,dc=shinken,dc=local"
- ajouter le modèle "Datacenter_01_Bordeaux" à la fin du champs use aux hôtes ou la propriété "location" vaut "FRBXDC01"
- ajouter le modèle "generic-contact" à tous les contacts découvert par la source
- ajouter le modèle "domain-admins" à tous les contact qui font partie du groupe ldap "cn=Domain Admins,ou=Users,dc=shinken,dc=local"

/etc/shinken-user/source-data/source-data-active-directory-sample/_configuration/openldap-rules.json

```
{
  "hosts_template": "linux",
  "hosts_template_centos": "OU=centos,OU=serveurs,dc=shinken,dc=local",
  "AddLast_template_(Datacenter_01_Bordeaux)_to_host_matching_[location]": "FRBXDC01",

  "contacts_template": "generic-contact",
  "AddFirst_template_(domain-admins)_to_contact_matching_[memberOf]": "cn=Domain Admins,ou=Users,dc=shinken,
dc=local",
}
```

Pour ne définir aucune règle vous pouvez créer un fichier vide avec seulement les accolades ouvrantes et fermantes ou ne pas préciser de fichier dans la définition de la source.

Import des objets

Pour importer des objets, allez sur la page d'accueil de l'**UI de configuration**, si votre configuration est bonne, vous devriez avoir un message "**OK: la source LDAP a été correctement chargée.**"

? Unknown Attachment

Maintenant, faîtes un "**Forcer l'import**" en cliquant sur

Dans le panneau "**Elements >**" vous verrez les nouveaux éléments apparaître (Hôtes et Contacts).

? Unknown Attachment

La prochaine étape sera alors d'importer ces nouveaux éléments dans Shinken.

HOW TO

Importer des ordinateurs avec des noms spécifiques

? Unknown Attachment	Dans le fichier **openldap-connection.json**

Modifiez le paramètre **hosts_filter**

```
  "hosts_filter": "(&(objectClass=computer)(sAMAccountName=*SERVER_NAME*))",
```

Changez **SERVER_NAME** par le nom de serveur vous voulez importer.

Importer des utilisateurs issus d'un ou plusieurs groupes

Avec la source OpenLDAP, il est donc possible d'importer des utilisateurs de la base contacts_base spécifiée dans le fichier **openldap-connection.json** mais on peut aussi les filtrer afin de n'importer que ceux qui sont dans un ou plusieurs groupes différents de l'annuaire LDAP.

? Unknown Attachment	Dans le fichier **openldap-connection.json**

Dans **contacts_filter_with_group,** ajouter le Distinguished Name (DN) des différents groupes d'utilisateurs séparés par un pipe ("|")

/etc/shinken-user/source-data/source-data-active-directory-sample/_configuration/active-directory-rules.json

```
  "contacts_filter_with_group": "CN=shinken_admins,OU=utilisateurs,DC=shinkendom,DC=local | CN=shinken_users,
OU=utilisateurs,DC=shinkendom,DC=local",
```

Filtrer et appliquer des modèles

Cette source inclut également d'autres paramètres qui permettent d'appliquer des modèles automatiquement suivant le type d'objets :

- ***hosts_template***: chaque hôte chargé aura au moins le modèle défini en valeur
- ***contacts_template***: chaque contact chargé aura au moins le modèle défini en valeur

Il est également possible d'ajouter un modèle sur les hôtes sur leur OU (Organization Unit) en utilisant le paramètre ***hosts_template_**

Par exemple, si vous voulez ajouter le modèle **exchange** à tous les serveurs qui sont dans l'OU: OU=Email Collaboration Servers,OU=DataCenter Servers,DC=YOUR,dc=DOMAIN,dc=com, utilisez le paramètre suivant:

```
"hosts_template_exchange":  "OU=Email Collaboration Servers,OU=DataCenter Servers,DC=YOUR,dc=DOMAIN,dc=com"
```

Précisions techniques

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 techniques sur le fonctionnement de l'import des sources.

Pour une source OpenLDAP, la propriété **properties_used_as_synckey** ne fonctionne pas. Il faut utiliser la propriété **properties_used_as_synckey_for_TYPE** (TYPE : le type de l'élément) qui permet de définir une clé de synchronisation pour chaque type d'élément.

Il y a 4 type d'éléments disponibles :

Type d'élément	Nom du type en cfg	Propriétés par défaut servant de clé
Hôte	host	host_name, address
Groupe d'hôte	hostgroups	hostgroup_name
Utilisateur	contacts	contact_name, email
Groupe d'utilisateur	contactgroups	contactgroup_name

Pour les éléments du type "utilisateur" peut avoir comme propriété servant de clé de synchronisation **short_mail**. Cette propriété correspond à la partie avant le caractère "@". Exemple: utilisateur1@shinken-solutions.com seul utilisateur1 sera prise en compte comme clé.