

## ANNEXE

<b><u>1]. Serveur DHCP</u></b> (via terminale, déconseillé)	<b><u>P.4</u></b>
Configuration du serveur DHCP pour servir d'autre réseau que le réseau VOIP auquel il est connecté.	<b><u>P.5</u></b>
<b><u>2]. CUSTOM TEMPLATE</u></b>	<b><u>P.6</u></b>
Creating custom templates	<b><u>P.6</u></b>
Gabarit personnalisé pour chaque appareil	<b><u>P.7</u></b>
Modèle personnalisé pour un modèle spécifique	<b><u>P.8</u></b>
Modèle personnalisé pour un appareil spécifique	<b><u>P.8</u></b>
Modification du plugin utilisé par un périphérique	<b><u>P.9</u></b>
<b><u>3]. Intégration de Berofos avec PBX</u></b>	<b><u>P.10</u></b>
<b><u>4]. Directory Xlet (Annuaire Xlet)</u></b>	<b><u>P.12</u></b>
Vue d'ensemble	<b><u>P.12</u></b>
Utilisation	<b><u>P.12</u></b>
Répertoire téléphonique	<b><u>P.12</u></b>
Configuration	<b><u>P.13</u></b>
Filtre d'affichage	<b><u>P.13</u></b>
Contexte et association des filtres	<b><u>P.14</u></b>
<b><u>5]. Musique d'attente</u></b>	<b><u>P.15</u></b>
<b><u>6]. fichier sauvegardé/non sauvegardé</u></b>	<b><u>P.16</u></b>
<b><u>7]. Tableau des flux réseau (IN)</u></b>	<b><u>P.18</u></b>
<b><u>8]. Xivo-Confd</u></b> (mécanisme d'organisation de plugin)	<b><u>P.19</u></b>
<b><u>9]. WAZO-DIRD SOURCE CONFIGURATION</u></b>	<b><u>P.21</u></b>
SCHEMA WAZO_DIRD fonctionnement	<b><u>P.22</u></b>
Back-end	<b><u>P.24</u></b>
<b><u>A].Détail d'Implémentation</u></b>	<b><u>P.24</u></b>
Service (plugin)	<b><u>P.25</u></b>
<b><u>B].Détail d'Implémentation</u></b>	<b><u>P.26</u></b>
View (plugin)	<b><u>P.26</u></b>
<b><u>C].Détail d'Implémentation</u></b>	<b><u>P.27</u></b>
<b><u>10]. XIVO purge-db</u></b> (gestion des plugins relatifs au dictionnaire)	<b><u>P.28</u></b>
<b><u>11]. Xivo-sysconfd</u></b> (configuration du serveur Wazo)	<b><u>P.29</u></b>
<b><u>12]. STOCK plugins Documentation</u></b>	<b><u>P.31</u></b>
View plugins (descriptif des plugin view)	<b><u>P.31</u></b>
<b><u>A]. SERVICE PLUGIN</u></b>	<b><u>P.31</u></b>

Configuration (dictionnaire ou les clés sont des noms de profil)	<u><b>P.32</b></u>
Template (génère le contenu du fichier configuration)	<u><b>P.32</b></u>
Host configuration	<u><b>P.33</b></u>
profile et service association	<u><b>P.34</b></u>
<b><u>B]. Back-end configuration</u></b>	<u><b>P.34</b></u>
CSV	<u><b>P.34</b></u>
CSV web service	<u><b>P.35</b></u>
dird_phonebook (cherche dans l'annuaire)	<u><b>P.35</b></u>
ldap (cherche des entrées dans le serveur LDAP)	<u><b>P.36</b></u>
Personal (cherche dans annuaire personnel)	<u><b>P.37</b></u>
Xivo (ajoute des utilisateurs dans l'annuaire via Wazo)	<u><b>P.38</b></u>
<b><u>13]. Consul</u></b> (découverte de service)	<u><b>P.39</b></u>
installation de consul	<u><b>P.39</b></u>
activer l'agent de configuration	<u><b>P.41</b></u>
<b><u>14]. DEVICES supportés</u></b>	<u><b>P.42</b></u>
Aastra	<u><b>P.43</b></u>
Alcatel-Lucent	<u><b>P.44</b></u>
Avaya	<u><b>P.44</b></u>
Cisco SPA 300&500	<u><b>P.44</b></u>
Cisco Small Business IP Phones	<u><b>P.44</b></u>
Cisco ATAs	<u><b>P.45</b></u>
Cisco 79XX	<u><b>P.46</b></u>
Digium	<u><b>P.49</b></u>
Fanvil	<u><b>P.50</b></u>
Gigaset	<u><b>P.50</b></u>
Jitsi	<u><b>P.50</b></u>
Patton	<u><b>P.50</b></u>
Panasonic	<u><b>P.51</b></u>
Polycom	<u><b>P.52</b></u>
Snom	<u><b>P.54</b></u>
Technicolor	<u><b>P.56</b></u>
Yealink	<u><b>P.57</b></u>
Zenitel	<u><b>P.58</b></u>
<b><u>15].Débogage Astérisk</u></b>	<u><b>P.59</b></u>
Il y a donc un problème avec Asterisk. Et maintenant quoi ?	<u><b>P.59</b></u>
Débogage d'un crash d'astérisk	<u><b>P.60</b></u>

Debug d'un freeze Asterisk	<b><u>P.60</u></b>
Recompiler Astérisk	<b><u>P.61</u></b>
Recompiler une version vanille d'Asterisk (Wazo < 17.17)	<b><u>P.61</u></b>
Recompiler une version vanille d'Asterisk (Wazo >= 17.17)	<b><u>P.62</u></b>
Exécuter Asterisk sous Valgrind	<b><u>P.63</u></b>
<b><u>16].Débogage des daemons</u></b>	<b><u>P.64</u></b>
xivo-confgend	<b><u>P.64</u></b>
Xivo-provd	<b><u>P.64</u></b>
Consul	<b><u>P.64</u></b>
Mongooseim	<b><u>P.65</u></b>
<b><u>17].Dépannage</u></b>	<b><u>P.66</u></b>
Transferts par DTMF	<b><u>P.66</u></b>
Détection de fax	<b><u>P.66</u></b>
Le serveur CTI se termine de manière inattendue	<b><u>P.67</u></b>
Agents recevant deux appels DAA	<b><u>P.68</u></b>
Erreurs de localisation PostgreSQL	<b><u>P.69</u></b>
Le cluster de base de données ne démarre pas	<b><u>P.69</u></b>
Impossible de se connecter à la base de données	<b><u>P.70</u></b>
Erreur lors de la mise à niveau	<b><u>P.70</u></b>
Erreur lors de la restauration d'une sauvegarde de base de données	<b><u>P.70</u></b>
Erreur lors de la réplication maître-esclave	<b><u>P.71</u></b>
Modification de la locale (LC_COLLATE et LC_CTYPE) de la base de données	<b><u>P.71</u></b>
Lancer un appel depuis la console Asterisk	<b><u>P.71</u></b>
Capture de paquets réseau	<b><u>P.72</u></b>
Obtenir de l'aide	<b><u>P.72</u></b>
Collecte des logs	<b><u>P.74</u></b>
<b><u>18].Synchroniser un appareil</u></b>	<b><u>P.75</u></b>
Synchroniser plusieurs appareils	<b><u>P.76</u></b>
<b><u>19].Liens externe</u></b>	<b><u>P.77</u></b>

## 1]. Serveur DHCP (via terminale, déconseillé)

pour ajouter un seul

pour changer la route par défaut du serveur dhcp. P73 (77/499) de la doc wazo.

Il faut créer un dossier pour le DHCP\_subnet.conf.head :

```
mkdir -p /etc/xivo/custom-templates/dhcp/etc/dhcp/  
#ensuite ce placer dans le répertoire.  
cd /etc/xivo/custom-templates/dhcp/etc/dhcp/  
#puis copier le fichier souhaité.  
cp /usr/share/xivo-config/templates/dhcp/etc/dhcp/dhcpd_subnet.conf.head .
```

Éditer le fichier avec vim ou nano.

```
vim dhcpd_subnet.conf.head
```

Dans le fichier, remplacer l'ensemble **#XIVO\_NET4\_IP#** par l'adresse de la route de notre réseau VOIP, par exemple :

```
option routers 192.168.2.254;
```

Re-générer la configuration dhcp :

```
xivo -update-config
```

Pour que le DHCP attribue des ip à toutes machines sur le réseau (comme des pc) :

Les OUIs et class de vendeurs (sur adresse mac) sont déclarés dans :

```
/etc/dhcp/dhcpd_update/* files
```

Si on veut que le serveur dhcp de wazo face du dhcp même sur les classes non connu (comme des pc) :

créer un dossier pour le fichier dhcpd\_subnet.conf.tail :

```
mkdir -p /etc/xivo/custom-templates/dhcp/etc/dhcp/
```

déplacement dans le dossier créer :

```
cd /etc/xivo/custom-templates/dhcp/etc/dhcp/
```

copie du fichier dans cet emplacement :

```
cp /usr/share/xivo-config/templates/dhcp/etc/dhcp/dhcpd_subnet.conf.tail
```

Éditer le fichier :

```
vim dhcpd_subnet.conf.tail
```

Ajouter la ligne suivante en tête du fichier :

```
allow unknown-clients;
```

re-généré la configuration du dhcp :

```
xivo-update-config
```

il ne reste plus qu'à redémarrer le serveur DHCP.

### **CONFIGURATION du serveur dhcp pour servir d'autre réseau que le réseau VOIP auquel il est connecté.**

Il faut créer un dossier et y copier deux fichiers car pour l'exemple nous allons créer deux réseaux :

```
mkdir /etc/dhcp/dhcpd_sites/  
cp /etc/dhcp/dhcpd_subnet.conf /etc/dhcp/dhcpd_sites/dhcpd_site001.conf  
cp /etc/dhcp/dhcpd_subnet.conf /etc/dhcp/dhcpd_sites/dhcpd_lanDATA.conf
```

on peut créer autant de site (fichier) dans dhcpd\_sites que l'on souhaite.

Après avoir créé le ou les fichier(s) dans `/etc/dhcp/dhcpd_sites` il faut éditer le fichier `/etc/dhcp/dhcpd_extra.conf` et ajouter l'emplacement des fichiers comme suivant :

```
include "/etc/dhcp/dhcpd_sites/dhcpd_siteXXX.conf";  
include "/etc/dhcp/dhcpd_sites/dhcpd_lanDATA.conf";
```

Ensuite il faut éditer chaque fichier de configuration dans `/dhcpd_sites/`.

changer les valeurs à changer :

```
subnet 172.30.8.0 netmask 255.255.255.0 { # indique le réseau et son masque  
option subnet-mask 255.255.255.0; #indique l'option du masque (envoyer au machine)  
option broadcast-address 172.30.8.255; #indique le broadcast, pas forcément présente ni  
#nécessaire  
option routers 172.30.8.1; #indique la route par défaut des machines  
dans la section pool changer le nom :  
pool {  
log(concat("[", binary-to-ascii(16, 8, ":", hardware), "] POOL VoIP Site XXX"));  
  
#range : définit la plage d'adresse ip à fournir.  
range 172.30.8.10 172.30.8.200;
```

il ne reste plus qu'à redémarrer le service DHCP :

service isc-dhcp-server restart

## 2]. CUSTOM TEMPLATE

### Creating custom templates :

Les modèles personnalisés s'avèrent pratiques lorsque vous avez une configuration vraiment spécifique à faire sur vos appareils téléphoniques.

Les modèles sont gérés par plugin. Il n'est pas possible qu'un template soit partagé par plus d'un plugin. C'est une limitation de conception du système de plugins de provd.



#### Note

Lorsque vous installez un nouveau plugin, les modèles ne sont pas migrés automatiquement, vous devez donc les copier manuellement de l'ancien répertoire du plugin vers le nouveau. Ceci ne s'applique pas à la mise à jour d'un plugin.

Supposons que nous ayons installé le plugin `xivo-aastra-3.3.1-SP2` et que nous voulions écrire des modèles personnalisés pour celui-ci.

La première chose à faire est d'aller dans le répertoire où le plugin est installé :

```
cd /var/lib/xivo-provd/plugins/xivo-aastra-3.3.1-SP2
```

Une fois que vous y êtes, vous pouvez voir qu'il y a pas mal de fichiers et de répertoires :

```
tree
.
+-- common.py
+-- entry.py
+-- pkgs
|   +-- pkgs.db
+-- plugin-info
+-- README
+-- templates
|   +-- 6730i.tpl
|   +-- 6731i.tpl
|   +-- 6739i.tpl
|   +-- 6753i.tpl
|   +-- 6755i.tpl
|   +-- 6757i.tpl
|   +-- 9143i.tpl
|   +-- 9480i.tpl
|   +-- base.tpl
+-- var
    +-- cache
    +-- installed
    +-- templates
    +-- tftpboot
        +-- Aastra
            +-- aastra.cfg
```

Les répertoires intéressants sont :

templates

C'est ici que se trouvent les modèles originaux. Vous ne devez pas éditer ces fichiers directement mais copier celui que vous voulez modifier dans le répertoire `var/templates`.

`var/templates`

C'est le répertoire dans lequel vous mettez et éditez vos modèles personnalisés.

`var/tftpboot`

C'est ici que se trouvent les fichiers de configuration une fois qu'ils ont été générés à partir des modèles. Vous devriez les consulter pour confirmer que vos modèles personnalisés vous donnent le résultat que vous attendez.



#### Avertissement

Lorsque vous désinstallez un plugin, le répertoire des plugins est supprimé, y compris tous les modèles personnalisés.

Quelques éléments à savoir avant d'écrire votre premier modèle personnalisé :

- ils utilisent le moteur de template Jinja2. (<http://jinja.pocoo.org/docs/2.10/templates/>)
- lors de la création d'un `include` ou d'un `extend` à partir d'un modèle, le fichier est d'abord recherché dans le répertoire `var/templates`, puis dans le répertoire `templates`.
- Les modèles sont affectés en mode autoprov, car du point de vue de `provd`, il n'y a pas de différence entre un périphérique en mode autoprov ou entièrement configuré. Cela signifie qu'il n'est généralement pas nécessaire de modifier les fichiers statiques dans `var/tftpboot`. Dans les faits ce serait une mauvaise idée car une mise à jour du plugin écraserait ces fichiers.

### **Gabarit personnalisé pour chaque appareil**

```
cp templates/base.tpl var/templates
vi var/templates/base.tpl
xivo-provd-cli -c 'devices.using_plugin("xivo-aastra-3.3.1-SP2").reconfigure()'
```

Une fois fait, si vous voulez synchroniser tous les périphériques affectés, utilisez la commande suivante :

```
xivo-provd-cli -c 'devices.using_plugin("xivo-aastra-3.3.1-SP2").synchronize()'
```

## Modèle personnalisé pour un modèle spécifique

Supposons que nous voulions personnaliser le modèle pour notre 6739i :

```
cp templates/6739i.tpl var/templates
vi var/templates/6739i.tpl
xivo-provd-cli -c 'devices.using_plugin("xivo-aastra-3.3.1-SP2").reconfigure()'
```

## Modèle personnalisé pour un appareil spécifique

Pour créer un modèle personnalisé pour un périphérique spécifique, vous devez créer un modèle spécifique au périphérique nommé `<device_specific_file_with_extension>.tpl` dans le répertoire `var/templates/` :

- pour un téléphone Aastra, si vous souhaitez personnaliser le fichier `00085D2EECFB.cfg`, vous devrez créer un fichier modèle nommé `00085D2EECFB.cfg.tpl`,
- pour un téléphone Snom, si vous voulez personnaliser le fichier `000413470411.xml`, vous devrez créer un fichier modèle nommé `000413470411.xml.tpl`,
- pour un téléphone Polycom, si vous souhaitez personnaliser le fichier `0004f2211c8b-user.cfg`, vous devrez créer un fichier modèle nommé `0004f2211c8b-user.cfg.tpl`,
- et ainsi de suite.

Ici, nous voulons personnaliser le contenu d'un fichier spécifique à l'appareil nommé `00085D2EECFB.cfg`, nous devons créer un modèle nommé `00085D2EECFB.cfg.tpl` :

```
cp templates/6739i.tpl var/templates/00085D2EECFB.cfg.tpl
vi var/templates/00085D2EECFB.cfg.tpl
xivo-provd-cli -c
'devices.using_mac("00085D2EECFB").reconfigure()'
```

### Note



Le choix d'utiliser cette syntaxe vient du fait que provd supporte des périphériques qui n'ont pas d'adresse MAC, à savoir les softphones.

De plus, certains appareils ont plus d'un fichier (comme Snom), et cette méthode permet de personnaliser plus d'un fichier.

Le template à utiliser comme base pour un template spécifique à un dispositif variera en fonction des besoins. Généralement, le modèle de template sera un bon choix, mais ce ne sera pas toujours le cas.



## **Modification du plugin utilisé par un périphérique**

De temps en temps, de nouveaux firmwares sont lancés par les fabricants d'appareils. Cela se traduit parfois par la disponibilité d'un nouveau plugin pour ces périphériques.

Quand ça se produit, cela signifie presque toujours que le nouveau plugin rend obsolète l'ancien. L'ancien plugin est alors considéré comme " en fin de vie ", et ne recevra pas de nouvelles mises à jour et, ne sera pas disponible pour une nouvelle installation.

Supposons que nous ayons l'ancien plugin `xivo-aastra-3.2.2.1136` installé sur notre Wazo et que nous voulons utiliser le nouveau plugin `xivo-aastra-3.3.1-SP2`.

Ces deux plugins peuvent être installés en même temps, et vous pouvez changer manuellement le plugin utilisé par un téléphone en le modifiant via la page Services › IPBX › Terminaisons.

Si vous utilisez des modèles personnalisés dans votre ancien plugin, vous devez les copier dans le nouveau plugin et vous assurer qu'ils sont toujours compatibles.

Une fois que vous avez pris la décision de migrer tous vos téléphones vers le nouveau plugin, vous pouvez utiliser la commande suivante :

```
xivo-provd-cli -c 'helpers.mass_update_devices_plugin("xivo-aastra-3.2.2.1136", "xivo-aastra-3.3.1-SP2")'
```

Ou, si vous voulez aussi les synchroniser (c'est-à-dire les redémarrer) en même temps :

```
xivo-provd-cli -c 'helpers.mass_update_devices_plugin("xivo-aastra-3.2.2.1136", "xivo-aastra-3.3.1-SP2", synchronize=True)'
```

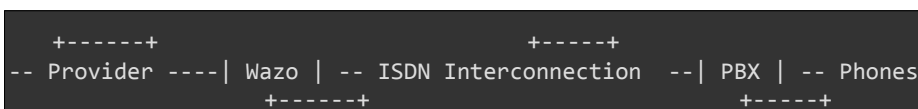
Vous pouvez vérifier que tout s'est bien passé en consultant la page Services › IPBX › IPBX › Terminaisons.

### 3]. Intégration de Berofos avec PBX

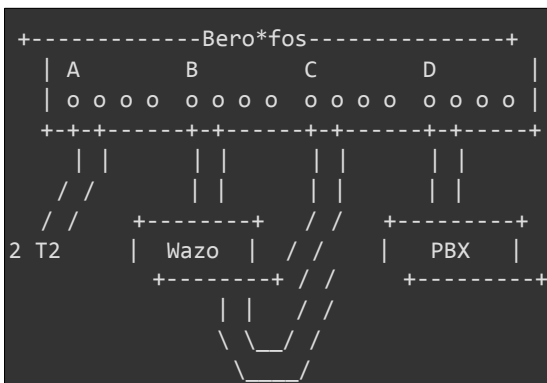
Vous pouvez utiliser un commutateur basculeur Berofos pour sécuriser les lignes du fournisseur ISDN lorsque vous installez un Wazo devant un PBX existant. Le but de cette configuration est d'atténuer les conséquences d'une panne du Wazo : avec cet équipement, les liaisons des fournisseurs RNIS pourraient être commutées directement sur le PBX en cas de panne de Wazo.

Wazo n'offre pas nativement la possibilité de configurer Berofos dans ce mode de basculement. Cette section décrit une solution de contournement.

Vue logique :



Connexion :



Ce qui suit décrit comment configurer votre Wazo et votre Berofos.

1.Suivez la configuration générale de Berofos (firmware, IP, login/mot de passe) décrite dans la page Installation et configuration de Berofos.

2.Une fois terminé, appliquez ces paramètres spécifiques au berofos :

```
bnfos --set scenario=1 -h 10.105.2.26 -u admin:berofos
bnfos --set mode=1 -h 10.105.2.26 -u admin:berofos
bnfos --set modedef=1 -h 10.105.2.26 -u admin:berofos
bnfos --set wdog=1 -h 10.105.2.26 -u admin:berofos
bnfos --set wdogdef=1 -h 10.105.2.26 -u admin:berofos
bnfos --set wdogitime=60 -h 10.105.2.26 -u admin:berofos
```

3. Ajoutez le script suivant `/usr/local/sbin/berofos-workaround` :

```
#!/bin/bash
# Script workaround for berofos integration with a Wazo in front of PABX

res=$(/usr/sbin/service asterisk status)
does_ast_run=$?
if [ $does_ast_run -eq 0 ]; then
    /usr/bin/logger "$0 - Asterisk is running"
    # If asterisk is running, we (re)enable wdog and (re)set the mode
    /usr/bin/bnfos --set mode=1 -f fos1 -s
    /usr/bin/bnfos --set modedef=1 -f fos1 -s
    /usr/bin/bnfos --set wdog=1 -f fos1 -s

    # Now 'kick' berofos ten times each 5 seconds
    for ((i = 1; i <= 10; i += 1)); do
        /usr/bin/bnfos --kick -f fos1 -s
        /bin/sleep 5
    done
else
    /usr/bin/logger "$0 - Asterisk is not running"
fi
```

4. Ajouter des droits d'exécution au script :

```
chmod +x /usr/local/sbin/berofos-workaround
```

5. Créez un cron pour lancer le script toutes les minutes `/etc/cron.d/berofos-cron-workaround` :

```
# Workaround to berofos integration
MAILTO=""

*/1 * * * * root /usr/local/sbin/berofos-workaround
```

## 4]. Directory XLET (Annuaire Xlet)

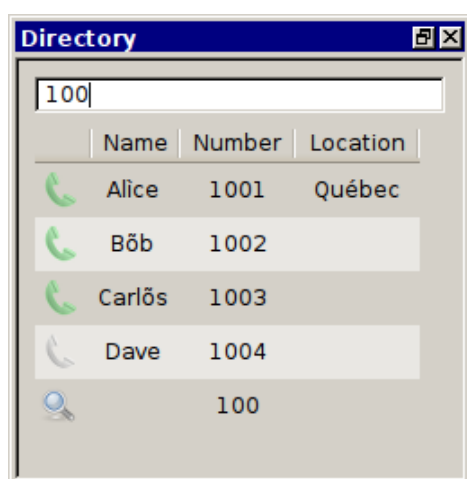
### Vue d'ensemble



#### Avertissement

Cet xlet ne doit être utilisée qu'avec un profil Switchboard. Il n'est pas destiné à être utilisé seul.

Le but de l'annuaire xlet est de permettre à l'utilisateur de rechercher parmi les utilisateurs Wazo, les entrées d'annuaires et les numéros arbitraires pour pouvoir appeler et transférer des appels vers ces destinations.



Légende		
Utilisateurs disponibles	Users available	
Utilisateurs qui sonnent	Users ringing	
Utilisateurs parlant	Users talking	
Utilisateurs	Users	
Téléphone portable	Mobile phone	
Contacts externes	External contacts	
Recherche actuelle (pas un contact)	Current search	

### Utilisation

La liste des entrées du xlet est recherchée dans le champ supérieur. Les entrées sont filtrées par contenu de colonne. La liste des entrées apparaîtra d'abord comme vide.

Si le terme de recherche actuel est un numéro valide, il sera affiché dans la liste des résultats sans nom pour permettre le transfert vers des numéros qui ne sont pas actuellement dans le répertoire ou configurés sur le Wazo.

### Répertoire téléphonique

Les recherches dans le répertoire téléphonique sont déclenchées dès que l'utilisateur a entré 3 caractères. Les résultats des répertoires distants apparaîtront après 1 seconde.

Si une entrée de répertoire a le même numéro qu'un mobile ou un téléphone configuré sur le Wazo, des colonnes supplémentaires seront ajoutées à l'entrée correspondante au lieu de créer une nouvelle ligne dans le résultat de la recherche.

Par exemple :

Si l'utilisateur 1 a le numéro 1000 et se trouve également dans un LDAP configuré avec un emplacement dans "Québec", et si le filtre d'affichage contient la colonne Emplacement(québec). l'entrée pour l'utilisateur 1 affichera "Québec" dans la colonne Emplacement une fois les résultats de la recherche reçus.

## Configuration

### Contexte

Le répertoire xlet a besoin d'un contexte spécial nommé `__switchboard_directory`. Dans Services > IPBX > IPBX > Configuration IPBX > contextes.

ajouter un nouveau contexte avec les paramètres suivants :

Nom : `__switchboard_directory`

Type de contexte : **Other**

Nom d'affichage : Switchboard

Name	Displayed name	Context type	Entity	Action
<input type="checkbox"/> > default	default	Internal	pcm-dev (pcm-dev)	
<input type="checkbox"/> > from-extern	Incalls	Incall	pcm-dev (pcm-dev)	
<input type="checkbox"/> > invalid	invalid	Incall	pcm-dev (pcm-dev)	
<input type="checkbox"/> > pcm-dev	pcm-dev	Internal	pcm-dev (pcm-dev)	
<input type="checkbox"/> > statscenter	statscenter	Internal	pcm-dev (pcm-dev)	
<input type="checkbox"/> > <b>__switchboard_directory</b>	Switchboard	<b>Other</b>	pcm-dev (pcm-dev)	
<input type="checkbox"/> > to-extern	Outcalls	Outcall	pcm-dev (pcm-dev)	

## Filtre d'affichage

Un nouveau filtre d'affichage doit être créé pour le répertoire xlet.

Update displays

Name:

Field title	Field type	Default value	Field name	
<input type="text"/>	<input type="text" value="status"/>	<input type="text"/>	<input type="text"/>	
<input type="text" value="Number"/>	<input type="text" value="number_office"/>	<input type="text"/>	<input type="text" value="number"/>	
<input type="text" value="Name"/>	<input type="text" value="name"/>	<input type="text"/>	<input type="text" value="name"/>	
<input type="text" value="Source"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="source"/>	
<input type="text" value="Number"/>	<input type="text" value="number_mobile"/>	<input type="text"/>	<input type="text" value="mobile"/>	

Description

You need to restart the Dird server to apply changes.

Save

Les champs suivants doivent être configurés avec la valeur correcte pour la colonne Type de champ(field type) afin que les entrées soient affichées dans le xlet :

- 1.status est la colonne qui sera utilisée pour afficher l'icône d'état, le titre peut être vide
- 2.Name est affiché dans la colonne Nom de la xlet
- 3.number\_office s'affiche dans la colonne Nombre avec une icône de téléphone dans la xlet
- 4.number\_mobile est affiché dans la colonne Nombre avec une icône mobile dans la xlet
- 5.number\_.... tout autre champ commençant par number\_ sera affiché dans la colonne Number de la xlet avec une icône de répertoire générique
- 6.Tout autre champ sera affiché dans leur propre colonne du répertoire xlet

Les valeurs de la colonne Field Name doivent contenir les valeurs créées dans la définition du répertoire.

Le titre utilisé pour la colonne Nombre est le titre du premier champ (field name) dont le type commence par nombre\_.



#### Note

Le titre du champ de la première colonne numérique sera utilisé pour le titre de l'en-tête dans la xlet.



#### Avertissement

Assurez-vous que les champs saisis dans le format d'affichage sont également disponibles dans la définition du répertoire, sinon le filtre ne retournera aucun résultat.

## **Contexte et association des filtres**

Le nouveau filtre d'affichage doit être affecté au contexte du répertoire  
\_\_switchboard\_directory

Vous pouvez ensuite choisir quels répertoires seront recherchés par le Xlet.



#### Avertissement

Vous ne devez pas sélectionner le répertoire interne, car il est déjà géré.

## 51. Musique d'attente.

Le menu Services ▸ IPBX ▸ IPBX ▸ services IPBX ▸ Musique d'attente mène à la liste des musiques d'attente disponibles.

Fichier de musiques, Formats supporté :

Format Name	Filename Extension
G.719	.g719
G.723	.g723 .g723sf
G.726	.g726-40 .g726-32 .g726-24 .g726-16
G.729	.g729
GSM	.gsm
iLBC	.ilbc
Ogg Vorbis	.ogg (only mono files sampled at 8000 Hz)
G.711 A-law	.alaw .al .alw
G.711 $\mu$ -law	.pcm .ulaw .ul .mu .ulw
G.722	.g722
Au	.au
Siren7	.siren7
Siren14	.siren14
SLN	.raw .sln .sln12 .sln16 .sln24 .sln32 .sln44 .sln48 .sln96 .sln192
VOX	.vox
WAV	.wav .wav16
WAV GSM	.WAV .wav49

Un seul canal audio doit être présent par fichier, c'est-à-dire que les fichiers doivent être en mono.

Si vos fichiers de musique d'attente ne semblent pas fonctionner, cherchez des erreurs dans les journaux d'astérisques. La musique d'attente sera toujours jouée dès le début.

### **mp3 : lecture de fichiers MP3.**



#### Avertissement

Le mode mp3 est obsolète et vous ne devez pas l'utiliser. Au lieu de cela, il faut convertir vos fichiers MP3 dans un autre format et utiliser le mode "fichiers".

La musique d'attente sera diffusée à partir d'une position arbitraire sur la piste, elle ne sera pas diffusée dès le début.

- ].custom : ne joue pas les fichiers son. Au lieu de cela, exécutez un processus externe. Ce processus doit envoyer sur stdout le même format binaire que les fichiers WAV.

Exemple de processus :

```
/usr/bin/mpg123 -s --mono -y -f 8192 -r 8000 http://streaming.example.com/stream.mp3
```



#### Note

Les processus exécutés par les catégories personnalisées sont lancés dès que la catégorie est créée, et ne s'arrêtent que lorsque la catégorie est supprimée. Cela signifie que la musique d'attente alimentée par le streaming en ligne recevra constamment du trafic réseau, même lorsqu'il n'y a pas d'appels.

## 6]. fichier sauvegardé/non sauvegardé.

### Fichiers sauvegardés

```
/etc/asterisk/  
•/etc/consul/  
•/etc/crontab  
•/etc/dahdi/  
•/etc/dhcp/  
•/etc/hostname  
•/etc/hosts  
•/etc/ldap/  
•/etc/mongooseim/  
•/etc/network/if-up.d/xivo-routes  
•/etc/network/interfaces  
•/etc/ntp.conf  
•/etc/profile.d/xivo_uuid.sh  
•/etc/resolv.conf  
•/etc/ssl/  
•/etc/systemd/  
•/etc/wanpipe/  
•/etc/wazo-admin-ui/  
•/etc/wazo-auth/  
•/etc/wazo-call-logd/  
•/etc/wazo-dird/  
•/etc/wazo-plugind/  
•/etc/wazo-webhookd/  
•/etc/xivo-agentd/  
•/etc/xivo-agid/  
•/etc/xivo-amid/  
•/etc/xivo-confd/  
•/etc/xivo-configend-client/  
•/etc/xivo-ctid/  
•/etc/xivo-ctid-ng/  
•/etc/xivo-dird-phoned/  
•/etc/xivo-dxtora/  
•/etc/xivo-purge-db/  
•/etc/xivo-websocketd/  
•/etc/xivo/  
•/usr/local/bin/  
•/usr/local/sbin/  
•/usr/share/xivo/XIVO-VERSION  
•/var/lib/asterisk/  
•/var/lib/consul/  
•/var/lib/xivo-provd/  
•/var/lib/xivo/  
•/var/log/asterisk/  
•/var/spool/asterisk/  
•/var/spool/cron/crontabs/
```



Les fichiers suivants sont **exclus** des sauvegardes :

```
/var/lib/consul/checks  
/var/lib/consul/raft  
/var/lib/consul/serf  
/var/lib/consul/services  
/var/lib/xivo-provd/plugins/*/var/cache/*  
/var/spool/asterisk/monitor  
/var/spool/asterisk/meetme/  
/var/lib/xivo-provd/plugins/xivo-polycom*/var/tftpboot/*.ld
```

Fichier de logs, coredump files (mémoire vive et registre processus), enregistrement audio.

Et tous les fichiers suivants s'ils contiennent plus de **100 lignes ou sont  $\geq$  10MiB** ne sont pas sauvegardés.

```
/var/lib/xivo/sounds/  
/var/lib/asterisk/sounds/custom/  
/var/lib/asterisk/moh/  
/var/spool/asterisk/voicemail/  
/var/spool/asterisk/monitor/
```

## 7]. Tableau des flux réseau (IN)

Daemon Name	Service	Protocol	Port	Listen	Authentication	Enabled
-	ICMP	ICMP	-	0.0.0.0	no	yes
postfix	SMTP	TCP	25	0.0.0.0	yes	yes
isc-dhcpd	DHCP	UDP	67	0.0.0.0	no	no
isc-dhcpd	DHCP	UDP	68	0.0.0.0	no	no
xivo-provd	TFTP	UDP	69	0.0.0.0	no	yes
ntpd	NTP	UDP	123	0.0.0.0	yes	yes
monit	HTTP	TCP	2812	127.0.0.1	no	yes
asterisk	SIP	UDP	5060	0.0.0.0	yes	yes
asterisk	IAX	UDP	4569	0.0.0.0	yes	yes
asterisk	SCCP	TCP	2000	0.0.0.0	yes	yes
asterisk	AMI	TCP	5038	127.0.0.1	yes	yes
asterisk	HTTP	TCP	5039	127.0.0.1	yes	yes
asterisk	HTTPS	TCP	5040	127.0.0.1	yes	yes
sshd	SSH	TCP	22	0.0.0.0	yes	yes
nginx	HTTP	TCP	80	0.0.0.0	yes	yes
nginx	HTTPS	TCP	443	0.0.0.0	yes	yes
munin	HTTP	TCP	4949	127.0.0.1	no	yes
xivo-ctid	XiVO-CTI/S	TCP	5003	0.0.0.0	yes	yes
postgresql	SQL	TCP	5432	127.0.0.1	yes	yes
rabbitMQ	AMQP	TCP	5672	0.0.0.0	yes	yes
consul	Consul RPC	TCP	8300	127.0.0.1	yes	yes
consul	Consul Serf LAN	TCP/UDP	8301	127.0.0.1	yes	yes
consul	Consul Serf WAN	TCP/UDP	8302	127.0.0.1	yes	yes
consul	Consul HTTPS	TCP	8500	127.0.0.1	both	yes
xivo-provd	HTTP	TCP	8666	127.0.0.1	no	yes
xivo-provd	HTTP	TCP	8667	0.0.0.0	no	yes
xivo-confgend	HTTP	TCP	8669	127.0.0.1	no	yes
xivo-sysconfd	HTTP	TCP	8668	127.0.0.1	no	yes
wazo-admin-ui	HTTPS	TCP	9296	0.0.0.0	yes	yes
wazo-auth	HTTPS	TCP	9497	0.0.0.0	both	yes
wazo-call-logd	HTTPS	TCP	9298	0.0.0.0	yes	yes
wazo-dird	HTTPS	TCP	9489	0.0.0.0	yes	yes
wazo-webhookd	HTTPS	TCP	9300	0.0.0.0	yes	yes
xivo-confd	HTTPS	TCP	9486	0.0.0.0	yes	yes
xivo-confd	HTTP	TCP	9487	127.0.0.1	no	yes
xivo-amid	HTTPS	TCP	9491	0.0.0.0	yes	yes
xivo-agentd	HTTPS	TCP	9493	0.0.0.0	yes	yes
xivo-ctid	HTTP	TCP	9495	127.0.0.1	no	yes
xivo-dird-phoned	HTTP	TCP	9498	0.0.0.0	IP filtering	yes
xivo-dird-phoned	HTTPS	TCP	9499	0.0.0.0	IP filtering	yes
xivo-ctid-ng	HTTPS	TCP	9500	0.0.0.0	yes	yes
xivo-websocketd	WSS	TCP	9502	0.0.0.0	yes	yes
wazo-plugind	HTTPS	TCP	9503	0.0.0.0	yes	yes

## 8]. Xivo-Conf

Xivo-Conf ressources est un mécanisme d'organisation de plugins.  
Il y a deux principales catégories de plugin.

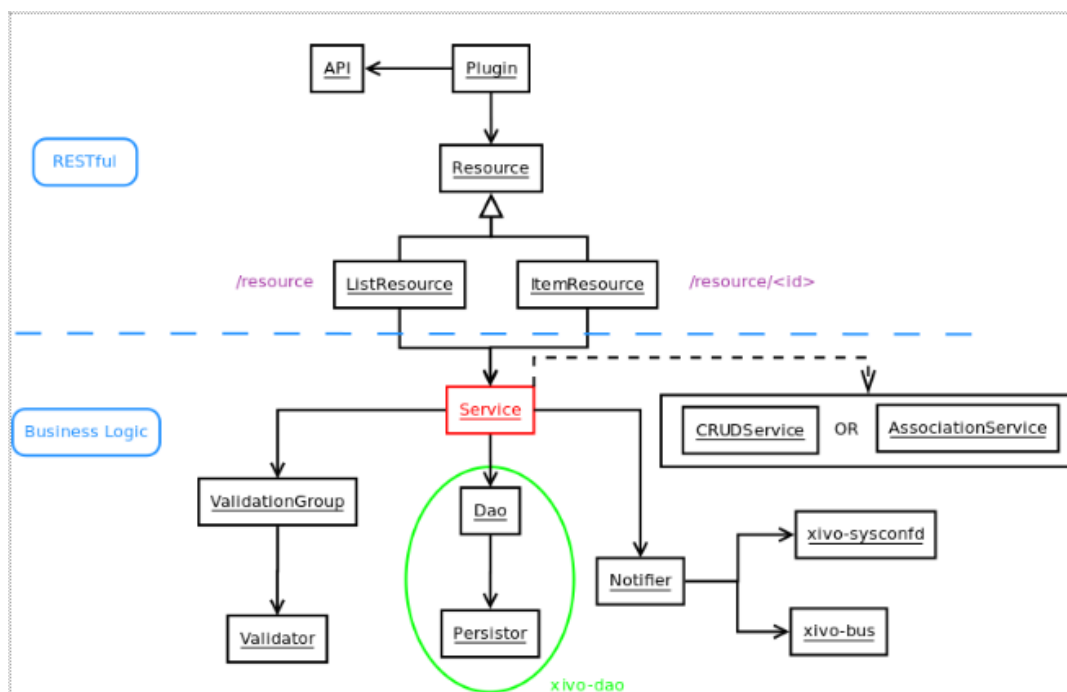
### Plugins de ressource :

un plugin qui manage des ressources. (ex : user ; extensions, voicemails etc..) un plugin de ressource utilise les 4 opérations Basic CRUD (Create, Read, Update, Delete) pour fonctionner sur une ressource à la manière d'un RESTful.

### Plugins d'associations :

un plugin pour associer ou dissocier deux ressources (ex un user et une Line).  
Il effectue en HTTP, soit une action d'association ( **POST** or **PUT** ), et une action de dissociation ( **DELETE** )

architecture de la gestion d'un plugin sous Xivo-Conf



### ressource :

classe qui traite les requêtes HTTP. La ressource utilise flash-restful pour gérer les requêtes.

Il y a deux types de ressources :

- ListResource pour URLs root, et ItemResource pour URLs qui possède ID.

- ListResource gère la création de ressource (POST) et chercher dans une liste de ressource disponible (GET).

ItemResource gère l'extraction d'un item (GET), la mise à jour (PUT), et la suppression (DELETE).

### Service :

Une classe qui assure la sous-traitance logique d'une ressource.

Notamment pour faire Get, Create, Update, ou supprimer une ressource. La classe service ne manipule pas directement les données, elle se contente de coordonner les actions via d'autres objets.

### Il y a deux types de services :

- CRUDService pour les opérations basiques CRUD dans les plugins de ressources.

- AssociationService pour associer ou dissocier des opérations dans le plugin association.

### Dao :

Data access object. sait prendre des données, comment les manipulées, pour des fichiers ou des requêtes SQL etc.

### Notifier :

Envoie des notifications d'évènements à chaque opération complétée. Le message sera ajouté à une file d'attente de message pour chaque opération CRUD effectué. Car certaines ressources doivent envoyer des évènements à d'autre daemon pour recharger certaine données de configuration. (ex : Asterisk à besoin de recharger le dialplan quand une extension est mise à jour.)

### Validator :

S'assure que la donnée de ressource ne contient aucune erreur avant de faire quoi que se soit avec.

Un validateur peut-être utilisé pour valider des data en entré, ou des règles administrative.

## 9]. WAZO-DIRD SOURCE CONFIGURATION

il y a deux possibilités pour configurer sources :

- dans la section source de la configuration principale. (config globale)
- dans un fichier d'un répertoire, un fichier pour une source (config ciblé)

emplacement par défaut : `/etc/wazo-dird/sources.d`

Tous les fichiers listés dans ce répertoire seront lu et utilisés pour créer une source de donnée pour wazo-dird.

Un exemple de configuration source CSV, contenu de son fichier.

```
type: csv
name: my_contacts_in_a_csv_file
file: /usr/local/share/my_contacts.csv
unique_column: id
searched_columns:
  - fn
  - ln
format_columns:
  name: "{fn} {ln}"
  number: "{num}"
```

c'est équivalent au fichier de configuration principale.

```
sources:
  my_contacts_in_a_csv_file:
    type: csv
    name: my_contacts_in_a_csv_file
    file: /usr/local/share/my_contacts.csv
    unique_column: id
    searched_columns:
      - fn
      - ln
    source_to_display_columns:
      ln: lastname
      fn: firstname
      num: number
```

### **type**

le type de la source, il doit être identique au nom du back-end (plugin) utilisé et activé.

### **name :**

c'est le nom de cette configuration. Le nom est utilisé pour associer la source au profil. La valeur est arbitraire, mais elle doit obligatoirement être unique.



Changer le nom de la source fera disparaître tous les favoris de la source. Il n'y a actuellement aucun outils pour aider à la migration de favoris entre deux noms de sources. Donc choisissez vos noms de source consciencieusement.

Les autres options sont dépendantes du type de source (le back-end utilisé, voir le stock plugins documentation). Toutefois, les clés suivantes doivent être présentes dans toutes les sources de configuration.

#### **first\_matched\_columns (optional)**

La colonne utilisée pour le reverse lookup. Toute colonne ayant le terme recherché renverra un résultat reverse lookup.

#### **format\_columns(optional)**

Un mapping entre un champ de résultat et une chaîne de format. La nouvelle clé sera ajoutée au résultat, si le nom existe déjà dans le résultat, il sera remplacé avec la nouvelle valeur. La syntaxe est une chaîne au format python.(voir <https://docs.python.org/2/library/string.html#formatspec> pour plus d'information).

#### **Searched\_columns(optional)**

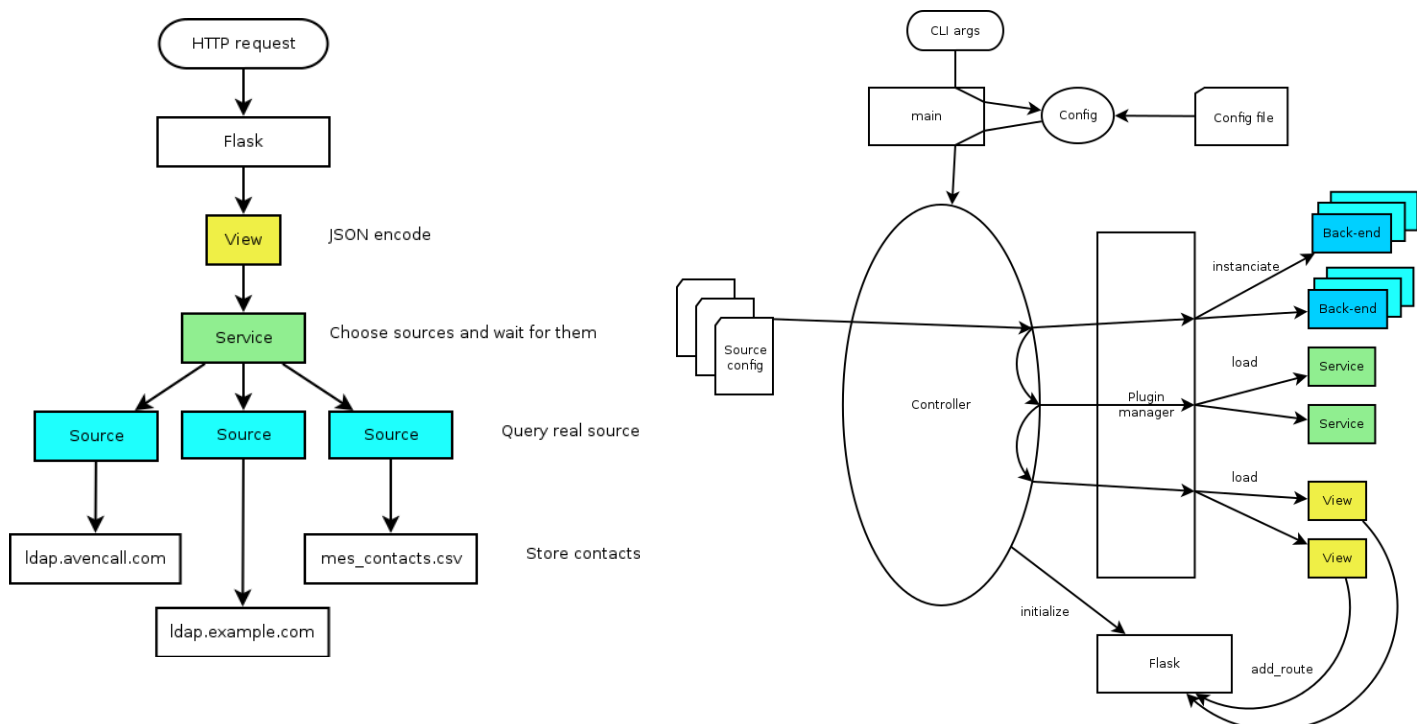
La colonne utilisée pour la recherche. Toutes colonnes contenant la chaîne de termes cherchée sera un résultat de recherche.

#### **unique\_columns(optional)**

Cette colonne est ce qui fait une entrée unique dans la source. **unique\_column** est utilisé pour construire l'**uid** qui est passé à la méthode **list** pour récupérer une liste de résultats par identifiant unique.

Elle est nécessaire pour l'inscription et l'identification des favoris.

### **SCHÉMA WAZO\_DIRD fonctionnement**



L'architecture wazo-dird utilise des plugins comme points d'extension pour la plupart de son travail. Il utilise **stevedore**\* pour faire l'instanciation et la découverte des plugins et des classes ABC pour définir l'interface requise.

Les plugins de wazo-dird utilisent les points d'entrée de setuptools\*. Cela signifie que l'installation d'un nouveau plugin sur wazo-dird nécessite un point d'entrée dans le fichier setup.py du plugin.

Chaque partie d'un plugin effectue des actions spécifiques.

Back-end		Service		View
----------	--	---------	--	------

Tous les plugins sont instanciés par le noyau. Le noyau conserve ensuite un catalogue des extensions chargées qui peuvent être fournies à d'autres extensions.

Le fichier **setup.py** suivant montre un exemple de bibliothèque python qui ajoute un plugin de chaque type à wazo-dird :

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

from setuptools import setup
from setuptools import find_packages

setup(
    name='Wazo dird plugin sample',
    version='0.0.1',

    description='An example program',

    packages=find_packages(),

    entry_points={
        'wazo_dird.services': [
            'my_service = dummy:DummyServicePlugin',
        ],
        'wazo_dird.backends': [
            'my_backend = dummy:DummyBackend',
        ],
        'wazo_dird.views': [
            'my_view = dummy:DummyView',
        ],
    }
)
```

\*stevedore :<https://docs.openstack.org/stevedore/latest/>

## **Back-end**

Les back-ends sont utilisés pour interroger les répertoires. Chaque back-end implémente un moyen d'interroger un répertoire donné. Chaque instance fournit par un back-end est appelée une source. Les sources sont utilisées par les services pour obtenir des résultats sur chaque répertoire configuré.

Par exemple grâce à un back-end LDAP, je peux configurer une source à partir d'un LDAP sur alpha.example.com et une seconde source à partir d'un second LDAP sur beta.example.com. Ces deux sources utilisent le même back-end LDAP.

### **A]. Détail d'Implémentation.**

-Namespace : `wazo_dird.backends`  
-Methods

`name` : le nom de la source. Récupéré à partir de la configuration injectée à `load()`

`load(args)` : configure les ressources utilisées par le plugin, en fonction de la config `args` qui est un dictionnaire contenant :

key `config` : la configuration source de cette instance du back-end

key `main_config` : Toute la config de wazo-dird.

`Unload()` : ressources gratuite utilisé par le plugin

`search(term, args)` : la méthode de recherche qui retourne une liste de dictionnaire.

`first_match(term, args)` : méthode qui retourne un dictionnaire.

Chaque valeur vide doit contenir `None` ,en lieu et place d'une chaîne vide.

`args` est un dictionnaire contenant :

key `token_infos` : association de donnée au token d'authentification. (voir wazo-auth)

`list(uids, args)` : la méthode liste retourne une liste de dictionnaires à partir d'une liste d'uids. Chaque uid est une chaîne de caractères identifiant un contact dans la source.



La mise en œuvre du back-end tient compte de ces valeurs et donnera les résultats correspondant.

Ex : on ajoute un backend qui renvoie des noms et des numéros aléatoires.

```
dummy.py :
# -*- coding: utf-8 -*-

import logging

logger = logging.getLogger(__name__)

class DummyBackendPlugin(object):

    def name(self):
        return 'my_local_dummy'

    def load(self, args):
        logger.info('dummy backend loaded')

    def unload(self):
        logger.info('dummy backend unloaded')

    def search(self, term, args):
        nb_results = random.randint(1, 20)
        return _random_list(nb_results)

    def list(self, unique_ids):
        return _random_list(len(unique_ids))

    def _random_list(self, nb_results):
        columns = ['Firstname', 'Lastname', 'Number']
        return [_random_entry(columns) for _ in xrange(nb_results)]

    def _random_entry(self, columns):
        random_stuff = [_random_string() for _ in xrange(len(columns))]
        return dict(zip(columns, random_stuff))

    def _random_string(self):
        return ''.join(random.choice(string.lowercase) for _ in xrange(5))
```

## **service (plugin)**

Les plugins de service ajoutent de nouvelles fonctionnalités au serveur Dird. Ces fonctionnalités sont disponibles pour les vues. Une fois chargé, un plugin de service reçoit sa configuration et un dictionnaire des sources disponibles.

Voici quelques exemples de services :

Un service de recherche pour rechercher dans toutes les sources configurées.

Un service de recherche inverse pour rechercher dans toutes les sources configurées et retourner un champ spécifique du premier résultat correspondant.

## **B]. Détail d'Implémentation.**

`load(args)` : configure les ressources utilisées par le plugin, en fonction de la configuration.

`args` est un dictionnaire contenant :

key `config` | key `sources`

`load` doit retourner l'objet de service, qui est n'importe quel type d'objet python.

`Unload()` : ressources gratuites utilisées par le plugin.

Ex : ajout d'un service qui retourne une liste vide lorsqu'il est utilisé.

```
dummy.py :
# -*- coding: utf-8 -*-

import logging
from wazo_dird import BaseServicePlugin
logger = logging.getLogger(__name__)
class DummyServicePlugin(BaseServicePlugin):
    """
    This plugin is responsible for instantiating and returning the
    DummyService. It manages its life time and should take care of
    its cleanup if necessary
    """
    def load(self, args):
        """
        Ignores all provided arguments and instantiate a DummyService that
        is returned to the core
        """
        logger.info('dummy loaded')
        self._service = DummyService()
        return self._service
    def unload(self):
        logger.info('dummy unloaded')
class DummyService(object):
    """
    A very dumb service that will return an empty list every time it is used
    """
    def list(self):
        """
        This function must be called explicitly from the view, `list` is not a
        special method name for wazo-dird
        """
        return []
```

## **View (plugin)**

Les plugins de visualisation ajoutent de nouvelles routes à l'application HTTP dans wazo-dird, en particulier l'API REST de wazo-dird : ils définissent les URL auxquelles wazo-dird va répondre et le formatage des données reçues et envoyées via ces URL.

Par exemple, nous pouvons définir une API REST formatée en JSON avec une vue et la même API formatée en XML avec une autre vue. La prise en charge de la fonction

répertoire d'un téléphone consiste généralement à ajouter une nouvelle vue pour le format que le téléphone consomme.

#### Détail d'Implémentation.

Namespace : `wazo_dird.views`

méthode :

`load(args)` : configure les ressources utilisées par le plugin, en fonction de la configuration. En général, enregistre les itinéraires sur Flask.

`args` est un dictionnaire contenant :

key `config` : la section du fichier de configuration pour toutes les vues sous forme de dict

key `services` : un dictionnaire des services, indexés par nom, qui peuvent être appelés par une route

key `http_app` : l'instance d'application Flask

key `rest_api` : un instance de Flask-RestFul Api

`Unload()` : ressources gratuites utilisées par le plugin.

## 10]. XIVO\_purge-db

Chaque plugin est un callable Python (constructeur de fonction ou de classe), qui prend un dictionnaire de configuration en argument. Les clés de ce dictionnaire sont les clés extraites du fichier de configuration. Ce qui vous permet d'ajouter une configuration spécifique au plugin dans `/etc/xivo-purge-db/conf.d/`.

Exemple de plugin dans le **xivo-purge-db git repo**. (<https://github.com/wazo-pbx/xivo-purge-db/tree/master/contribs>)

archive name : sample.

Objet: démonstration => comment créer votre propre plugin d'archive

activation du plugin

chaque plugin à besoin d'être explicitement activé dans la configuration de

`/tmp/xivo_purge_db.sample` avec le contenu suivant :

```
enabled_plugins:
  archives:
    - sample
```

### **sample.py**

L'exemple suivant fera une sauvegarde dans un fichier au format `/tmp/xivo_purge_db.sample` avec le contenu :

Save tables before purge. 365 days to keep!

**sample\_file = '/tmp/xivo\_purge\_db.sample'**

```
def sample_plugin(config):
    with open(sample_file, 'w') as output:
        output.write('Save tables before purge. {0} days to
keep!'.format(config['days_to_keep']))
```

installation du plugin sample :

Le **setup.py** montre un exemple d'une bibliothèque python qui ajoute un plugin à xivo-purge-db :

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

from setuptools import setup
from setuptools import find_packages

setup(
    name='xivo-purge-db-sample-plugin',
    version='0.0.1',

    description='An example program',
    packages=find_packages(),
    entry_points={
        'xivo_purge_db.archives': [
            'sample = xivo_purge_db_sample.sample:sample_plugin',
```

```
    },  
  }  
)
```

## 11]. Xivo-sysconfd

Xivo-sysconfd est le système de configuration du serveur Wazo. Il fait plusieurs choses différentes : (liste non exhaustive)

- configuration des interfaces réseau (interface, hostname, DNS)
- configuration de la haute disponibilité
- démarrage, arrêt, redémarrage des services
- rechargement de la configuration d'Asterisk
- envoi de certain évènement au composant (xivo-agentd, xivo-agid et xivo ctid)

### fichier de configuration

emplacement par défaut : `/etc/xivo/sysconfd.conf` format : INI

L'emplacement par défaut peut être écrasé par les options de ligne de commande.

Ex de fichier de configuration :

```
[general]  
xivo_config_path = /etc/xivo  
templates_path = /usr/share/xivo-sysconfd/templates  
custom_templates_path = /etc/xivo/sysconfd/custom-templates  
backup_path = /var/backups/xivo-sysconfd  
  
[resolvconf]  
hostname_file = /etc/hostname  
hostname_update_cmd = /etc/init.d/hostname.sh start  
hosts_file = /etc/hosts  
resolvconf_file = /etc/resolv.conf  
  
[network]  
interfaces_file = /etc/network/interfaces  
  
[wizard]  
templates_path = /usr/share/xivo-config/templates  
custom_templates_path = /etc/xivo/custom-templates  
  
[commonconf]  
commonconf_file = /etc/xivo/common.conf  
commonconf_cmd = /usr/sbin/xivo-update-config  
commonconf_monit = /usr/sbin/xivo-monitoring-update  
  
[openssl]  
certsdir = /var/lib/xivo/certificates  
  
[monit]  
monit_checks_dir = /usr/share/xivo-monitoring/checks  
monit_conf_dir = /etc/monit/conf.d
```

```
[request_handlers]
synchronous = true

[bus]
username = guest
password = guest
host = localhost
port = 5672
exchange_name = xivo
exchange_type = topic
exchange_durable = true
```

### Request handlers section

#### synchronisation

Si cette option est vraie, lorsque xivo-sysconfd reçoit par exemple une demande de rechargement du dialplan, il attendra que le rechargement du dialplan soit terminé pour répondre à la demande.

Lorsque cette option est fausse, xivo-sysconfd répond immédiatement à la requête.

Configurer cette option sur false accélérera certaines opérations (par exemple, éditer un utilisateur depuis l'interface web ou depuis xivo-confd), mais cela signifie qu'il y aura un petit délai (jusqu'à quelques secondes dans le pire des cas) entre le moment où vous créez votre utilisateur et celui où vous pouvez composer son extension avec succès.

## **12]. STOCK plugins Documentation.**

### **View plugins :**

#### **default\_json**

objet : présenter les entrées de répertoire au format JSON.

#### **Headers**

objet : liste les entête qui seront disponibles dans l'affichage de **default\_json**

#### **personal\_view**

objet : utilise l'API REST pour gérer les contacts personnels (cresta, delete, list).

#### **phonebook\_view**

objet : utilise l'API REST pour gérer les annuaires interne de wazo-dird's

#### **aastra\_view**

objet : utilise l'API REST pour chercher un téléphone Aastra dans le répertoire configuré

#### **cisco\_view**

objet : utilise l'API REST pour chercher un téléphone Cisco dans le répertoire configuré

#### **polycom\_view**

objet : utilise l'API REST pour chercher un téléphone polycom dans le répertoire configuré

#### **snom\_view**

objet : utilise l'API REST pour chercher un téléphone snom dans le répertoire configuré

#### **thomson\_view**

objet : utilise l'API REST pour chercher un téléphone thomson dans le répertoire configuré

#### **yealink\_view**

objet : utilise l'API REST pour chercher un téléphone Yealink dans le répertoire configuré

### **A]. SERVICE PLUGIN**

#### **-lookup**

objet: cherche dans des bases de donnée multiple, cherche des entrées correspondant à un mot.

## **Configuration** (extrait du fichier de configuration principal)

```
services:
  lookup:
    default:
      sources:
        my_csv: true
      timeout: 0.5
```

deux clés => sources : le nom de la source qui doit être utilisé. Timeout : l'attente maximum pour la réponse, au-delà du temps imparti la réponse est ignoré.

la configuration est un dictionnaire ou les clés sont des noms de profil et les valeurs les configuration spécifique de ces profils.

### **-favorite**

objet: marque/supprime les contacts comme favoris, et obtenir la liste de tous les favoris.

### **-personal**

objet:ajoute, supprime, affiche les contacts personnels d'un utilisateur.

### **-phonebook**

objet : ajoute, supprime, affiche les annuaires téléphoniques et les contacts de l'annuaire téléphonique.

### **-reverse = lookup**

objet :la configuration est un dictionnaire ou les clés sont des noms de profil et les valeurs les configuration spécifique de ces profils.

### **-service\_discovery**

objet : créer des sources lorsque les services sont enregistrés/autorisé avec la découverte de services.

Pour configurer de nouvelles sources le service nécessite les point suivants :

1. un template pour le fichier source de la configuration
2. un ensemble de configuration qui sera appliqué au modèle.
3. un ensemble de service et profil qui sera utilisé avec la nouvelle source.



La découverte d'un service est limité à un seul à la fois. Cela signifie que la découverte d'un serveur xivo-confd suppose que wazo-auth réside sur le même hôte ou que le template est déjà configuré avec le nom d'hôte approprié.

## **Template**

le template est utilisé pour générer le contenu du fichier de configuration pour le nouveau service. Il doit correspondre aux sources utilisées par un même back-end. Exemple 1



l'emplacement du template est configuré dans le service de configuration. Exemple 2

Ex1 :

```
type: xivo
name: wazo-{{ uuid }}
searched_columns:
- firstname
- lastname
first_matched_columns:
- exten
auth:
  host: {{ hostname }}
  port: 9497
  username: {{ service_id }}
  password: {{ service_key }}
  verify_certificate: false
confd:
  host: {{ hostname }}
  port: {{ port }}
  version: "1.1"
  verify_certificate: false
format_columns:
  name: "{{firstname}} {{lastname}}"
  phone: "{{exten}}"
  number: "{{exten}}"
  reverse: "{{firstname}} {{lastname}}"
  voicemail: "{{voicemail_number}}"
```

**uuid** : c'est l'uuid qui est dans la notification de registre.

**Hostname** : l'hôte annoncé depuis le service distant

**port** : le port annoncé depuis le service distant

**service\_id** : le login utilisé pour interroger xivo-confd

**service\_key** : la clé utilisée pour interroger xivo-confd

Ex2:

```
services:
  service_discovery:
    template_path: /etc/wazo-dird/templates.d
    services:
      xivo-confd:
        template: confd.yml
```

dans cet exemple le fichier `/etc/wazo-dird/templates.d` sera utilisé pour créer la configuration d'une nouvelle source quand un nouveau service xivo-confd est enregistré.

## **Host configuration**

La section host permet à l'administrateur de configurer certaines informations qui ne sont pas disponibles dans la découverte de services, mais qui le seront dans les modèles.

Il s'agit généralement de l'**ID\_service** et de la **key\_service** qui sont configurés avec l'ACL appropriée sur le Wazo distant.

Exemple:

```
services:
  service_discovery:
    hosts:
      ff791b0e-3d28-4b4d-bb90-2724c0a248cb:
        uuid: ff791b0e-3d28-4b4d-bb90-2724c0a248cb
        service_id: some-service-name
        service_key: secre7
        datacenter: dc1
        token: 3f031816-84a6-3960-fcd1-9cca67eacde2
```

### **profile et service association**

l'association entre le service et le profil pour la découverte sont définies dans la configuration du service `service_discovery`.

Exemple:

```
services:
  service_discovery:
    services:
      xivo-confd:
        lookup:
          default: true
          foobar: true
        reverse:
          foobar: true
        favorites:
          default: true
          foobar: true
```

Dans cet exemple, un nouveau service `xivo-confd` génère une configuration basée sur son modèle et cette nouvelle source est ajoutée pour la recherche et les favoris.

## **B]. Back-end configuration**

### **CSV**

- back-end name :`csv`
- objet : lire les entrées de répertoire d'un fichier `csv`.
- Limitation : le délimiteur `csv` n'est pas configurable (currently : `,` (comma))
- config : exemple (un fichier dans `source_config_dir` )

```
type: csv
name: my_csv
file: /var/tmp/test.csv
unique_column: id
searched_columns:
  - fn
  - ln
first_matched_columns:
  - num
```

```
format_columns:
  lastname: "{ln}"
  firstname: "{fn}"
  number: "{num}"
```

avec le fichier csv :

```
id,fn,ln,num
1,Alice,Abrams,55553783147
2,Bob,Benito,5551354958
3,Charles,Curie,5553132479
```

## **CSV web service**

- back-end name : csv\_ws
- Objet : lance une recherche à l'aide d'un service Web qui retourne des résultats au format CSV.

Sur l'exemple suivant, wazo-dird va appeler "<https://example.com:8000/ws-phonebook?firstname=alice&lastname=alice>" pour une recherche sur le terme "Alice"

### configuration :

exemple (sur un fichier dans `source_config_dir` )

```
type: csv_ws
name: a_csv_web_service
lookup_url: "https://example.com:8000/ws-phonebook"
list_url: "https://example.com:8000/ws-phonebook"
verify_certificate: False
searched_columns:
  - firstname
  - lastname
first_matched_columns:
  - exten
delimiter: ","
timeout: 16
unique_column: id
format_columns:
  number: "{exten}"
```

**lookup\_url** : l'url utilisé pour les recherches dans les répertoires.

**list\_url (optional)** : l'url utilisé pour lister les entrées disponibles. Elle est utilisée pour récupérer les favoris

**verify\_certificate (optional)** : dans le cas où le certificat est vérifié. Un chemin CA\_BUNDLE peut être fournie. Vrai par défaut.

**Delimiter (optional)** : délimiteur par défaut dans le résultat csv. Par défaut ','

**Timeout (optional)** : le nombre de seconde avant que la recherche sur le service web soit abandonnée. Valeur par défaut à vrai.

## **dird\_phonebook**

backend name : dird\_phonebook

Objet : recherche dans les annuaires interne de wazo-dird.

### Configuration : ex

```
type: dird_phonebook
name: phonebook
db_uri: 'postgresql://asterisk:proformatique@localhost/asterisk'
tenant: default
phonebook_id: 42
phonebook_name: main
first_matched_columns:
  - number
searched_columns:
  - firstname
  - lastname
format_columns:
  name: "{firstname} {lastname}"
```

**db\_uri** : L'URI de la base de donnée utilisé par wazo-dird pour stocker les annuaires.

**phonebook\_name** : le nom de l'annuaire utilisé pour cette ressource.

**phonebook\_id (obsolète, utilisé phonebook\_name)** : l'id au lieu du nom

## ldap

back-end name : ldap

objet : cherche des entrées d'annuaire à partir d'un serveur ldap.

Configuration : (ex d'un fichier dans `source_config_dir` )

```
type: ldap
name: my_ldap
ldap_uri: ldap://example.org
ldap_base_dn: ou=people,dc=example,dc=org
ldap_username: cn=admin,dc=example,dc=org
ldap_password: foobar
ldap_custom_filter: (l=québec)
unique_column: entryUUID
searched_columns:
  - cn
first_matched_columns:
  - telephoneNumber
format_columns:
  firstname: "{givenName}"
  lastname: "{sn}"
  number: "{telephoneNumber}"
```

**ldap\_uri** : l'URI du serveur ldap. Ne peut contenir que la partie schéma, host et port d'un URL LDAP.

**ldap\_base\_dn** : Le DN de l'entrée ou la recherche doit commencer.

**ldap\_username (optional)** : le DN de l'utilisateur à utiliser lors de l'exécution d'une liaison "simple". Par défaut il s'agit d'une chaîne vide.

Lorsque `ldap_username` et `ldap_password` sont vides, une liaison anonyme est effectuée.

**`ldap_password (optional)`** : le mot de passe utilisé lors de l'exécution d'une liaison simple. Par défaut c'est une chaîne vide.

**`ldap_custom_filter(optional)`** : le filtre personnalisé est utilisé pour ajouter plus de critères au filtre généré par le backend.

Ex :

```
ldap_custom_filter:(l=québec)
searched_columns:[cn,st]
```

entraînera l'utilisation du filtre suivant pour la recherche :

```
(&(l=québec)(|(cn=%Q*)(st=%Q*)))
```

Si seul le filtre personnalisé doit être utilisé, laissez le champ `searched_columns` vide.

Pour un filtre LDAP VALIDE : `(&(o=ACMÉ)(cn=%Q*))`

**`ldap_network_timeout(optional)`** : le temps maximum en seconde, qu'une opération LDAP par le réseau peut-prendre. Au-delà, il n'y aura pas de retour de résultat. Par défaut il est à 0.3

**`ldap_timeout(optional)`** : le temps maximum en seconde, qu'une opération LDAP peut prendre. Au-delà, il n'y aura pas de retour de résultat. Par défaut il est à 1.0

**`unique_column(optional)`** : la colonne qui contient un identificateur de l'entrée unique. Celui-ci est nécessaire pour l'inscription et l'identification des favoris.

Pour OpenLDAP, vous devez définir cette option sur "`entryUUID`".

Pour Active Directory, vous devez définir cette option sur "`objectGUID`" et également définir l'option "`unique_column_format`" sur "`binary_uuid`".

**`unique_column_format(optional)`** : le type de l'unique colonne renvoyée par le serveur LDAP qui a été interrogé. Les valeurs valides sont "`string`" ou "`binary_uuid`".

La valeur par défaut est "`string`".

## **Personal**

back-end name : personal

objet : rechercher des entrées d'annuaire parmi les contacts personnels des utilisateurs

Vous ne devriez avoir qu'une seule source de type `personal`, car une seule sera utilisée pour lister les contacts personnels. Le backend `personal` a besoin d'une installation de Consul fonctionnelle. Ce backend fonctionne avec le service personnel, qui permet aux utilisateurs d'ajouter des contacts personnels.

### **Configuration :**

```
#ex( un fichier dans source_config_dir )
type: personal
name: personal
```

```

first_matched_columns:
  - number
format_columns:
  firstname: "{firstname}"
  lastname: "{lastname}"
  number: "{number}"

```

`unique_column` n'est pas configurable, ça valeur est toujours `id` .

## **Xivo**

back-end name : xivo

objet : ajouter des utilisateurs à partir d'un Wazo (peut être distant) comme entrées de répertoire

Ce back-end requière un username et un password qui ont les permissions suffisantes pour lister les utilisateurs et obtenir les informations du serveur xivo-confd.

### **Required ACL**

confd.user.read

confd.infos.read

```

#configuration ex ( un fichier dans source_config_dir )
type: xivo
name: my_xivo
auth:
  host: xivo.example.com
  port: 9497
  username: admin
  password: password
  backend: xivo_service
  verify_certificate: "/usr/share/xivo-certs/server.crt"
confd:
  host: xivo.example.com
  port: 9486
  version: 1.1
  timeout: 3
  verify_certificate: "/usr/share/xivo-certs/server.crt"
unique_column: id
first_matched_columns:
  - exten
searched_columns:
  - firstname
  - lastname
format_columns:
  number: "{exten}"
  mobile: "{mobile_phone_number}"

```

#### **auth:host**

le hostname du service wazo-auth

#### **auth:port**

le port du service wazo-auth

#### **auth:username**

le username utilisé pour effectuer une requête sur xivo-confd pour rechercher des utilisateurs.

#### **Confid:host**

le hostname du service xivo\_confid

**confd:port**

le port du service xivo\_confd (par défaut 9496)

**confd:version**

la version de xivo-confd API.

## 13]. Consul

Wazo utilise consul pour la découverte de services. Lorsqu'un daemon est démarré, il s'enregistre sur le nœud consul configuré.

Consul modèle peut être utilisé pour générer les fichiers de configuration de chaque daemon qui nécessite la disponibilité d'un autre service. Consul modèle peut également être utilisé pour recharger le service approprié.

L'installation par défaut ce fait dans `/etc/consul/xivo/*.json` tous les fichiers sont installés par le package, et ne doivent en aucun cas être modifié par l'admin.

Celui-ci devra apporter ces modifications de configuration en ajoutant des fichiers dans `/etc/systemd.system/consul.service.d`

à l'installation est généré un token dans `/var/lib/consul/master_token`

Il est possible de faire tourner consul sur un autre poste et que le service consul sur le local soit en lecture seul.

Pour cela :

### **installation de consul :**

32 bits :

```
wget --no-check-certificate  
https://releases.hashicorp.com/consul/0.5.2/consul_0.5.2_linux_386.zip
```

64 bits :

```
wget --no-check-certificate  
https://releases.hashicorp.com/consul/0.5.2/consul_0.5.2_linux_amd64.zip
```

installation :

```
unzip consul_0.5.2_linux_386.zip
```

ou

```
unzip consul_0.5.2_linux_amd64.zip
```

```
mv consul /usr/bin/consul  
mkdir -p /etc/consul/xivo  
mkdir -p /var/lib/consul  
adduser --quiet --system --group --no-create-home \
```

```
--home /var/lib/consul consul
```

copier la configuration de consul venant de wazo sur un nouveau poste :  
dans le nouvel hôte consul, modifier `/etc/consul/xivo/config.json` et inclure les lignes suivantes :

```
"bind_addr": "0.0.0.0",  
"client_addr": "0.0.0.0",  
"advertise_addr": "<consul-host>"
```

Puis

```
# on the consul host  
scp root@<wazo-host>:/lib/systemd/system/consul.service /lib/systemd/system  
systemctl daemon-reload  
scp -r root@<wazo-host>:/etc/consul /etc  
scp -r root@<wazo-host>:/usr/share/xivo-certs /usr/share  
consul agent -data-dir /var/lib/consul -config-dir /etc/consul/xivo/
```

Pour pouvoir charger consul sur le systemd, il faut changer le groupe et l'utilisateur.  
Créer le fichier : `/etc/consul/agent/config.json`  
et le remplir avec :

```
{  
  "acl_datacenter": "<node_name>",  
  "datacenter": "xivo",  
  "server": false,  
  "bind_addr": "0.0.0.0",  
  "advertise_addr": "<wazo_address>",  
  "client_addr": "127.0.0.1",  
  "bootstrap": false,  
  "rejoin_after_leave": true,  
  "data_dir": "/var/lib/consul",  
  "enable_syslog": true,  
  "disable_update_check": true,  
  "log_level": "INFO",  
  "ports": {  
    "dns": -1,  
    "http": -1,  
    "https": 8500  
  },  
  "retry_join": [  
    "<remote_host>"  
  ],  
  "cert_file": "/usr/share/xivo-  
certs/server.crt",  
  "key_file": "/usr/share/xivo-certs/server.key"  
}
```

`node_name` : nom au choix pour ce nœud

`wazo_address` : adresse ip du serveur wazo

`remote_host` : l'adresse ip du nouveau consul. Attention si pare-feu que celui-ci ne bloque pas les échanges.



Ce fichier doit être accessible par l'utilisateur consul :

```
chown -R consul:consul /etc/consul/agent
```

**activer l'agent de configuration**

ajouter ou modifier : `/etc/systemd/system/consul.service.d/custom.conf`

```
[Service]
Environment=CONFIG_DIR=/etc/consul/agent
```

redémarrer le service consul

```
service consul restart
```

update de la section consul de xivo-ctid  
ajouter un fichier

```
/etc/xivo-ctid/conf.d/remote_consul.yml
```

contenant :

```
rest_api:
  http:
    listen: 0.0.0.0

service_discovery:
  advertise_address: <xivo-ctid-host>
  check_url: http://<xivo-ctid-host>:9495/0.1/infos
```

puis  
pour atteindre xivo-stid

```
xivo-ctid-host
```

## **14]. DEVICES supportés**

### **Provisioning :**

L'appareil est-il pris en charge par le système d'auto-alimentation ?

### **H-A**

L'appareil est-il supporté par le système de haute disponibilité ?

### **Directory XIVO**

L'appareil est-il supporté par le répertoire distant ? En d'autres termes, est-il possible de consulter le répertoire distant du XIVO depuis l'appareil ?

### **Funckeys**

Combien de touches de fonction peuvent être configurées sur l'appareil à partir de l'interface Web Wazo ?

Le nombre de touches de fonction configurables sur un appareil n'est pas nécessairement le même que le nombre de touches de fonction physiques dont dispose l'appareil.

Par exemple, un Aastra 6757i dispose de 12 touches physiques mais vous pouvez configurer 30 touches de fonction grâce au système de page.

### **Légende :**

Y =Yes / Supported

N = No / Not supported

NT =Not tested

NYT =Not yet tested

FK = Funckey

SK = SoftKey

HK = HardKey

MN = Menu

Les touches de fonction fonctionnent à l'aide des extensions dans Services ▸ Extensions. Il est important d'activer les touches de fonction que vous voulez utiliser.

De plus, l'option d'activation du transfert dans l'onglet Services de configuration utilisateur doit être activée pour utiliser les touches de fonction de transfert.

**1]** (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18) Testé signifie que le dispositif a été testé par l'équipe de développement Wazo et que les développeurs ont accès à ce dispositif.

**2]** (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18) Fkeys est le nombre de touches de fonction programmables que vous pouvez configurer depuis l'interface Web Wazo. Il ne correspond pas nécessairement au nombre de touches de fonction physiques de l'appareil (par exemple, un 6757i possède 12 touches physiques mais vous pouvez configurer 30 touches de fonction grâce au

système de page).

**31** (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18) Wazo HA signifie que l'appareil fonctionne avec Wazo HA.

## **Aastra**

Aastra a été acquise par Mitel en 2014. A Wazo, les téléphones des séries 6700 et 6800 sont toujours référencés comme téléphones Aastra, pour des raisons historiques et de compatibilité.

### **6700i & 6800i séries**

	6731i	6735i	6737i	6739i	6755i	6757i	6863i	6865i	6867i	6869i
Provisioning	Y	Y	Y	Y	Y	Y	Y	Y	Y	NT
H-A	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Directory XIVO	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Funckeys	8	26	30	55	26	30	0	8	38	68
<b>Supported programmable keys</b>										
User with supervision function	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Group	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Queue	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Conference Room with supervision function	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>General Functions</b>										
Online call recording	N	N	N	N	N	N	N	Y	Y	Y
Phone status	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Sound recording	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Call recording	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Incoming call filtering	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Do not disturb	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Group interception	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Listen to online calls	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Directory access	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Filtering Boss - Secretary	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>Transfers Functions</b>										
Blind transfer	HK	Y	Y	HK	Y	Y	HK	HK	HK	HK
Indirect transfer	HK	Y	Y	HK	Y	Y	HK	HK	HK	HK
<b>Forwards Functions</b>										
Disable all forwarding	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Enable/Disable forwarding on no answer	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Enable/Disable forwarding on busy	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Enable/Disable forwarding unconditional	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>Voicemail Functions</b>										
Enable voicemail with supervision function	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Reach the voicemail	Y	Y	Y	HK	Y	Y	N	Y	Y	Y
Delete messages from voicemail	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>Agent Functions</b>										
Connect/Disconnect a static agent	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Connect a static agent	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Disconnect a static agent	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>Parking Functions</b>										
Parking	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
Parking position	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
<b>Paging Functions</b>										
Paging	Y	Y	Y	Y	Y	Y	N	Y	Y	Y

Modules Astra d'extension supportés :

Aastra® M670i (for Aastra® 35i/37i/39i/53i/55i/57i)  
Aastra® M675i (for Aastra® 35i/37i/39i/55i/57i)  
Aastra® M680 (for Aastra® 6865i/6867i/6869i)  
Aastra® M685 (for Aastra® 6865i/6867i/6869i)

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
6730i	No	8	Yes
6753i	Yes	6	Yes
6757i	Yes	30	Yes
9143i	Yes	7	Yes
9480i	No	6	Yes
9480CT	No	6	Yes

## **Alcatel-Lucent**

IP Touch séries

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
4008 Extended Edition	Yes	4	No
4018 Extended Edition	Yes	4	No

Notez que vous ne devez pas télécharger le micrologiciel pour ces téléphones à moins que vous n'acceptiez qu'il vienne d'une source non officielle.

Pour que le plugin fonctionne pleinement, vous avez besoin de ces paquets supplémentaires :

```
apt-get install p7zip python-pexpect telnet
```

## **Avaya**

1200 series IP Deskphones (auparavant connus sous le nom de Nortel IP Phones)

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
1220 IP	Yes	0	No
1230 IP	No	0	No

## **Cisco**

Cisco Small Business SPA300 series.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
SPA301	No	1	No
SPA303	No	3	No



Les touches de fonction sont partagées avec les touches de ligne pour tous les téléphones SPA.

## **Cisco Small Business SPA500 series**

Le module d'extension SPA500 est pris en charge.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
SPA501G	Yes	8	No
SPA502G	No	1	No
SPA504G	Yes	4	No
SPA508G	Yes	8	No
SPA509G	No	12	No
SPA512G	No	1	No
SPA514G	No	4	No
SPA525G	Yes	5	No
SPA525G2	No	5	No

## Cisco Small Business IP Phones (previously known as Linksys IP Phones)

### Note



Vous devez installer le firmware de chaque téléphone SPA9xx que vous utilisez car il redémarre en boucle lorsqu'il ne trouve pas son firmware.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
SPA901	No	1	No
SPA921	No	1	No
SPA922	No	1	No
SPA941	No	4	No
SPA942	Yes	4	No
SPA962	Yes	6	No

Le module d'extension SPA932 est pris en charge.

## Cisco ATAs

Pour de meilleurs résultats, activez l'intégration DHCP sur votre Wazo.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
PAP2	No	0	No
SPA2102	No	0	No
SPA8800	No	0	No
SPA112	No	0	No

### Note



Ces appareils peuvent être utilisés pour connecter des télécopieurs. Pour un meilleur succès avec les fax, certains paramètres doivent être modifiés. Vous pouvez lire la section Using analog gateways (FAX).

### Note



Si vous souhaitez resynchroniser manuellement la configuration à partir de l'appareil ATA, vous devez utiliser l'url suivante :

[http://ATA\\_IP/admin/resync?http://WAZO\\_IP:8667/CONF\\_FILE](http://ATA_IP/admin/resync?http://WAZO_IP:8667/CONF_FILE)

**ATA\_IP** est l'adresse IP de l'ATA,

**WAZO\_IP** est l'adresse IP de votre Wazo,

**CONF\_FILE** est l'un des fichiers spa2102.cfg, spa8000.cfg.

Pour de meilleurs résultats, activez l'intégration DHCP sur votre Wazo.

	SPA122	SPA3102	SPA8000
Provisioning	Y	Y	Y
H-A	N	N	N
Directory XIVO	N	N	N
Funckeys	0	0	0

Ces appareils peuvent être utilisés pour connecter des fax. Pour un meilleur succès

avec les fax, certains paramètres doivent être modifiés. Vous pouvez lire la section (FAX).

Pour resynchroniser voir note juste au-dessus.

## Cisco 79XX

	7905 G	7906 G	7911 G	7912 G	7920	7921 G	7940 G	7941 G	7941 G-GE	7942 G	7960 G	7961 G	7962 G
Provisioning	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
H-A	Y	Y	Y	Y	NT	NT	Y	Y	Y	Y	Y	Y	Y
Directory XIVO	FK	FK	FK	FK	N	N	FK	FK	FK	FK	FK	FK	FK
Funckeys	0	0	0	0	0	0	1	1	1	1	5	5	5
<b>Supported programmable keys</b>													
User with supervision function	N	N	N	N	N	N	N	N	N	N	N	N	N
Group	N	N	N	N	N	N	N	N	N	N	N	N	N
Queue	N	N	N	N	N	N	N	N	N	N	N	N	N
Conference Room with supervision function	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>General Functions</b>													
Online call recording	N	N	N	N	N	N	N	N	N	N	N	N	N
Phone status	N	N	N	N	N	N	N	N	N	N	N	N	N
Sound recording	N	N	N	N	N	N	N	N	N	N	N	N	N
Call recording	N	N	N	N	N	N	N	N	N	N	N	N	N
Incoming call filtering	N	N	N	N	N	N	N	N	N	N	N	N	N
Do not disturb	SK	SK	SK	SK	N	N	SK	SK	SK	SK	SK	SK	SK
Group interception	N	N	N	N	N	N	N	N	N	N	N	N	N
Listen to online calls	N	N	N	N	N	N	N	N	N	N	N	N	N
Directory access	Y	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y
Filtering Boss - Secretary	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>Transfers Functions</b>													
Blind transfer	N	N	N	N	N	N	N	N	N	N	N	N	N
Indirect transfer	SK	SK	SK	SK	SK	SK	SK	SK	SK	SK	SK	SK	SK
<b>Forwards Functions</b>													
Disable all forwarding	N	N	N	N	N	N	N	N	N	N	N	N	N
Enable/Disable forwarding on no answer	N	N	N	N	N	N	N	N	N	N	N	N	N
Enable/Disable forwarding on busy	N	N	N	N	N	N	N	N	N	N	N	N	N
Enable/Disable forwarding unconditional	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>Voicemail Functions</b>													
Enable voicemail with supervision function	N	N	N	N	N	N	N	N	N	N	N	N	N
Reach the voicemail	SK	SK	SK	SK	N	N	HK	HK	HK	HK	HK	HK	HK
Delete messages from voicemail	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>Agent Functions</b>													
Connect/Disconnect a static agent	N	N	N	N	N	N	N	N	N	N	N	N	N
Connect a static agent	N	N	N	N	N	N	N	N	N	N	N	N	N
Disconnect a static agent	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>Parking Functions</b>													
Parking	N	N	N	N	N	N	N	N	N	N	N	N	N
Parking position	N	N	N	N	N	N	N	N	N	N	N	N	N
<b>Paging Functions</b>													
Paging	N	N	N	N	N	N	N	N	N	N	N	N	N



### Avertissement

Ces téléphones ne peuvent être utilisés qu'en mode SCCP. Ils sont limités aux fonctionnalités prises en charge dans l'implémentation SCCP de Wazo.

Pour installer le firmware pour les plugins xivo-cisco-sccp, vous devez télécharger manuellement les fichiers du firmware depuis le site Web Cisco et les enregistrer dans le répertoire `/var/lib/xivo-provd/plugins/$plugin-name/var/cache`

Les permissions des fichiers doivent être modifiées pour rendre les fichiers lisibles par xivo-provd :

```
chmod 640 <nom de fichier>> chmod 640  
chown xivo-provd:xivo-provd <nom de fichier>>chown
```

Ce répertoire est créé par Wazo lorsque vous installez le plugin (i.e. xivo-cisco-sccp-legacy). Si vous créez le répertoire manuellement, l'installation échouera.



### Avertissement

L'accès aux mises à jour du firmware Cisco nécessite un compte Cisco avec suffisamment de privilèges. Le compte nécessite le paiement du service et reste sous la responsabilité du client ou du partenaire. Les auteurs de Wazo ne sont pas responsables de ces firmwares et ne proposent aucune mise à jour.

Par exemple, si vous avez installé le plugin `xivo-cisco-sccp-legacy` et que vous voulez installer le paquet `7940-7960-fw`, `networklocale` et `userlocale_fr_FR`, vous devez :

Rendez-vous sur <http://www.cisco.com>

Cliquez sur "Se connecter" dans le coin supérieur droit de la page, puis connectez-vous.

Cliquez sur le menu "Support

Cliquez sur l'onglet "Téléchargements", puis sur "Communications vocales et unifiées".

Sélectionnez "Téléphonie IP", puis "Unified Communications Endpoints", puis le modèle de votre téléphone (dans cet exemple, le 7940G).

Cliquez sur "Skinny Client Control Protocol (SCCP) software".

Choisissez la même version que celle montrée dans le plugin

Téléchargez le fichier avec une extension se terminant par ".zip", qui est généralement le dernier fichier de la liste.

Dans l'interface web de Wazo, vous pourrez ensuite cliquer sur le bouton "installer" du firmware.

La procédure est similaire pour la locale réseau et le paquet de locale utilisateur, mais :

Au lieu de cliquer sur "Skinny Client Control Protocol (SCCP) software", cliquez sur "Unified Communications Manager Endpoints Locale Installer".

Cliquez sur "Linux".


Choisissez la même version que celle montrée dans le plugin

Pour la locale réseau, téléchargez le fichier "po-locale-combined-network.cop.cop.sgn".

Pour la locale utilisateur, téléchargez le fichier nommé "po-locale-\$locale-\$locale-name.cop.sgn, par exemple "po-locale-fr\_FR.cop.sgn" pour la locale "fr\_FR".

Les deux fichiers doivent être placés dans le répertoire `/var/lib/xivo-provd/plugins/`  
`$plugin-name/var/cache` . Puis installez-les dans l'interface Web de Wazo.





**Note**  
Certaines touches de fonction sont partagées avec les touches de ligne

Particularités :

- Pour de meilleurs résultats, activez l'intégration DHCP sur votre Wazo.
- Impossible d'effectuer une prise de vue dirigée à l'aide d'une touche de fonction BLF.
- Prise en charge du DTMF uniquement en mode RFC2833.
- Ne fonctionne pas de manière fiable avec le commutateur PoE Cisco ESW520. Lorsqu'il est connecté à un tel commutateur, le D40 a tendance à redémarrer aléatoirement, et le D70 ne démarre pas du tout.
- Il est important de ne pas modifier la configuration du téléphone via l'interface web du téléphone lorsque vous utilisez ces téléphones avec Wazo.
- Le biper ne marche pas.

	D40	D50	D70
Provisioning	Y	NYT	Y
H-A	Y	NYT	Y
Directory XIVO	N	NYT	N
Funckeys	2	14	106
<b>Supported programmable keys</b>			
User with supervision function	N	NYT	N
Group	Y	NYT	Y
Queue	Y	NYT	Y
Conference Room with supervision function	Y	NYT	Y
<b>General Functions</b>			
Online call recording	N	NYT	N
Phone status	Y	NYT	Y
Sound recording	Y	NYT	Y
Call recording	Y	NYT	Y
Incoming call filtering	Y	NYT	Y
Do not disturb	HK	NYT	HK
Group interception	Y	NYT	Y
Listen to online calls	N	NYT	N
Directory access	N	NYT	N
Filtering Boss - Secretary	Y	NYT	Y
<b>Transfers Functions</b>			
Blind transfer	HK	NYT	HK
Indirect transfer	HK	NYT	HK
<b>Forwards Functions</b>			
Disable all forwarding	Y	NYT	Y
Enable/Disable forwarding on no answer	Y	NYT	Y
Enable/Disable forwarding on busy	Y	NYT	Y
Enable/Disable forwarding unconditional	Y	NYT	Y
<b>Voicemail Functions</b>			
Enable voicemail with supervision function	Y	NYT	Y
Reach the voicemail	HK	NYT	HK
Delete messages from voicemail	Y	NYT	Y
<b>Agent Functions</b>			
Connect/Disconnect a static agent	Y	NYT	Y
Connect a static agent	Y	NYT	Y
Disconnect a static agent	Y	NYT	Y
<b>Parking Functions</b>			
Parking	N	NYT	N
Parking position	N	NYT	N
<b>Paging Functions</b>			
Paging	Y	NYT	Y

## **Fanvil**

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
C62P	Yes	5	Yes

---

## **Gigaset**

Aussi connu sous le nom de Siemens.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
C470 IP	No	0	No
C475 IP	No	0	No
C590 IP	No	0	No
C595 IP	No	0	No
C610 IP	No	0	No
C610A IP	No	0	No
S675 IP	No	0	No
S685 IP	No	0	No
N300 IP	No	0	No
N300A IP	No	0	No
N510 IP PRO	No	0	No

---

## **Jitsi**

Les téléphones SIP des séries Mitel 6700 et 6800 sont pris en charge par Wazo. Voir la section Aastra.

---

## **Patton**

	SN4112	SN4114	SN4116	SN4118	SN4316	SN4324	SN4332
Provisioning	Y	Y	Y	Y	Y	Y	Y
H-A	Y	Y	Y	Y	Y	Y	Y

Wazo ne supporte que la configuration des ports FXS de ces passerelles. Il ne prend pas en charge la configuration des ports FXO.

Si vous avez une passerelle sur laquelle vous souhaitez configurer les ports FXO, vous devrez écrire manuellement la configuration des ports FXO en créant un modèle personnalisé pour votre passerelle.

Il n'est possible d'entrer un code de provisionnement que sur le premier port FXS d'une passerelle.

Par exemple, si vous avez une passerelle avec 8 ports FXS, le premier port peut être configuré en composant un code de provisioning à partir de celui-ci, mais les ports 2 à 7 ne peuvent être configurés que via l'interface Web Wazo.

De plus, si vous composez l'extension "reset to autoprov" depuis n'importe quel port, la configuration de tous les ports sera réinitialisée, pas seulement le port sur lequel l'extension a été appelée.

[Ces limites pourraient disparaître à l'avenir.](#)

Ces passerelles sont configurées avec quelques paramètres régionaux (France par défaut). Ces paramètres sont faciles à modifier en écrivant un modèle personnalisé.

L'accès Telnet et l'accès Web sont activés par défaut. Vous devez changer le mot de passe par défaut en définissant un mot de passe administrateur via un "template device" Wazo.

En téléchargeant et installant les firmwares Patton, vous acceptez les conditions de Patton Electronics Company.

Pour provisionner une passerelle précédemment configurée manuellement, utilisez les commandes suivantes sur votre passerelle (mode configuration), en remplaçant WAZO\_IP par l'adresse IP de votre serveur Wazo :

```
profile provisioning PF_PROVISIONING_CONFIG
  destination configuration
  location 1 http://WAZO_IP:8667/${system.mac}.cfg
  activation reload graceful
exit
provisioning execute PF_PROVISIONING_CONFIG
```

---

## **Panasonic**

Panasonic KX-HTXXX series:

Ces téléphones sont en test pour le moment

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
KX-HT113	No	—	No
KX-HT123	No	—	No
KX-HT133	No	—	No
KX-HT136	No	—	No

	Sound Point IP					Sound Station IP				Business media phone							
	SPIP3 31	SPIP3 35	SPIP4 50	SPIP5 50	SPIP5 60	SPIP6 50	SPIP5 000	SPIP6 000	SPIP7 000	VVX1 01	VVX2 01	VVX3 00	VVX3 10	VVX4 00	VVX4 10	VVX5 00	VVX6 00
Provisioning	NT	Y	Y	Y	NT	NT	NT	Y	NT	Y	Y	Y	Y	Y	Y	Y	NYT
H-A	N	Y	N	Y	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	N
Directory XIVO	N	N	N	FK	N	N	N	N	N	N	N	FK	FK	FK	FK	FK	N
Funkeys	N	0	2	3	3	47	0	0	0	0	0	6	6	12	12	12	0
<b>Supported programmable keys</b>																	
User with supervision function	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Group	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Queue	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Conference Room with supervision function	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>General Functions</b>																	
Online call recording	NYT	N	NYT	N	NYT	NYT	NYT	NYT	NYT	N	N	N	N	N	N	N	NYT
Phone status	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Sound recording	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Call recording	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Incoming call filtering	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Do not disturb	NYT	SK	NYT	HK	NYT	NYT	NYT	NYT	NYT	SK	SK	SK	SK	SK	SK	SK	NYT
Group interception	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Listen to online calls	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Directory access	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Filtering Boss - Secretary	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>Transfers Functions</b>																	
Blind transfer	NYT	SK	NYT	N	NYT	NYT	NYT	NYT	NYT	SK	SK	HK	HK	HK	HK	SK	NYT
Indirect transfer	NYT	SK	NYT	HK	NYT	NYT	NYT	NYT	NYT	SK	SK	HK	HK	HK	HK	SK	NYT
<b>Forwards Functions</b>																	
Disable all forwarding	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Enable/Disable forwarding on no answer	NYT	SK	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Enable/Disable forwarding on busy	NYT	SK	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Enable/Disable forwarding unconditional	NYT	SK	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>Voicemail Functions</b>																	
Enable voicemail with supervision function	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Reach the voicemail	NYT	SK	NYT	HK	NYT	NYT	NYT	NYT	NYT	SK	SK	HK	HK	HK	HK	SK	NYT
Delete messages from voicemail	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>Agent Functions</b>																	
Connect/Disconnect a static agent	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Connect a static agent	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Disconnect a static agent	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>Parking Functions</b>																	
Parking	NYT	N	NYT	N	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
Parking position	NYT	N	NYT	N	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT
<b>Paging Functions</b>																	
Paging	NYT	N	NYT	Y	NYT	NYT	NYT	NYT	NYT	Y	Y	Y	Y	Y	Y	Y	NYT

## Particularités :

- Les derniers firmwares Polycom peuvent prendre beaucoup de temps à télécharger et à installer en raison de leur taille (~650 MiB). Pour cette raison, ces fichiers sont explicitement exclus des sauvegardes Wazo.
- Pour que la prise d'appel dirigée fonctionne via les touches de fonction BLF, vous devez vous assurer que l'option **Set caller-id in dialog-info+xml notify** est activée sur votre Wazo. Cette option se trouve sur la page Services › IPBX › Paramètres généraux › Protocole SIP, dans l'onglet **Signaling**.

De plus, la prise d'appel dirigée via une touche de fonction BLF ne fonctionnera pas si le numéro de poste de l'utilisateur surveillé est différent de son numéro d'identification d'appelant.

- Le mot de passe par défaut est 9486 (c'est-à-dire le mot "xivo" sur le clavier d'un téléphone).
- Sur les VVX101 et VVX201, pour que les deux touches de ligne soient affectées à la même ligne SIP, créez un modèle personnalisé (custom templates) avec le contenu suivant :

```
{% extends 'base.tpl' -%}

{% block remote_phonebook -%}
{% endblock -%}

{% block model_specific_parameters -%}
reg.1.lineKeys="2"
{% endblock -%}
```

C'est particulièrement utile sur le VVX101, car il supporte un maximum de 1 ligne SIP et ne supporte pas les touches de fonction.



#### Note

(Wazo HA cluster) Les touches de fonction BLF enregistrées sur le nœud maître ne sont pas disponibles.

Modules d'extension supportés :

- Polycom® VVX Color Expansion (pour Polycom® VVX 300/310/400/400/410/500/600)
- Extension papier Polycom® VVX (pour Polycom® VVX 300/310/400/400/410/500/600)
- Polycom® SoundPoint IP rétroéclairé (pour Polycom® SoundPoint 650)

#### Avertissement

Les caméras Polycom® VVX® ne sont pas prises en charge.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
SPIP320	No	0	No
SPIP321	No	0	No
SPIP330	No	0	No
SPIP430	No	0	No
SPIP501	Yes	0	No
SPIP600	No	0	No
SPIP601	No	0	No
SPIP670	No	47	No
SPIP4000	No	0	No
VVX1500	No	0	No

## **Snom**

### **Avertissement**

Si vous utilisez des téléphones Snom avec HA, vous ne devez pas affecter plusieurs lignes au même appareil.

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
300	No	6	Yes
320	Yes	12	Yes
360	No	—	Yes
820	Yes	4	Yes
MP	No	—	Yes
PA1	No	0	Yes

Il y a un problème connu avec l'approvisionnement des téléphones Snom dans Wazo :

Après une réinitialisation d'usine d'un téléphone, si aucune langue et aucun fuseau horaire ne sont définis pour le "périphérique de configuration par défaut" dans Wazo › Configuration › Provisioning › Template device ›, vous serez forcé de sélectionner une langue et un fuseau horaire par défaut sur l'interface du téléphone.

### **Modules d'extension supportés :**

- Snom® Vision (pour les séries Snom® 7xx et Snom® 8xx)
- Snom® D7 (pour la série Snom® 7xx)

### **Note**

Pour certains modèles, les touches de fonction sont partagées avec les touches de ligne

Il y a les limitations/problèmes connus suivants concernant l'approvisionnement des téléphones Snom dans Wazo :

- Si vous utilisez des téléphones Snom avec HA, vous ne devez pas affecter plusieurs lignes au même appareil.
- Le Snom D745 dispose d'un espace limité pour les étiquettes des touches de fonction : les étiquettes longues peuvent être divisées de manière sous-optimale.
- Lors de l'utilisation d'un module d'extension D7, l'étiquette de la touche de fonction ne sera pas affichée lors du premier redémarrage ou de la resynchronisation. Vous devrez redémarrer ou resynchroniser le téléphone une deuxième fois pour que l'étiquette s'affiche correctement.
- Après une réinitialisation d'usine d'un téléphone, si aucune langue et aucun fuseau horaire ne sont définis pour le "périphérique de configuration par défaut" dans Wazo › Configuration › Provisioning › Template device ›, vous serez forcé de sélectionner une langue et un fuseau horaire par défaut sur l'interface du téléphone.

	370	710	715	720	D725	D745	760	D765	821	870
Provisioning	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
H-A	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Directory XIVO	HK	SK	SK	HK	HK	HK	HK	HK	HK	HK
Funckeys	12	5	5	18	18	32	16	16	12	15
<b>Supported programmable keys</b>										
User with supervision function	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Group	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Queue	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Conference Room with supervision function	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
<b>General Functions</b>										
Online call recording	N	N	N	N	N	N	N	N	N	N
Phone status	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Sound recording	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Call recording	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Incoming call filtering	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Do not disturb	HK	SK	SK	HK	HK	HK	HK	HK	HK	HK
Group interception	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Listen to online calls	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Directory access	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Filtering Boss - Secretary	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
<b>Transfers Functions</b>										
Blind transfer	Y	SK	SK	HK	HK	HK	HK	HK	HK	HK
Indirect transfer	Y	SK	SK	HK	HK	HK	HK	HK	HK	HK
<b>Forwards Functions</b>										
Disable all forwarding	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Enable/Disable forwarding on no answer	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Enable/Disable forwarding on busy	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Enable/Disable forwarding unconditional	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
<b>Voicemail Functions</b>										
Enable voicemail with supervision function	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Reach the voicemail	HK	HK	HK	HK	HK	HK	HK	HK	HK	HK
Delete messages from voicemail	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
<b>Agent Functions</b>										
Connect/Disconnect a static agent	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Connect a static agent	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Disconnect a static agent	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
<b>Parking Functions</b>										
Parking	Y	N	N	N	N	N	N	N	Y	Y
Parking position	Y	N	N	N	N	N	N	N	Y	Y
<b>Paging Functions</b>										
Paging	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

## **Tecnhicolor**

Anciennement connu sous le nom de Thomson

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
ST2022	No	—	—
ST2030	Yes	10	Yes

### Note

Les touches de fonction sont partagées avec les touches de ligne.

---



## Yealink

	T19 P	T19P E2	T20 P	T21 P	T21P E2	T22 P	T26 P	T28 P	T32G	T38G	T40P	T41P	T42 G	T46 G	T48 G	W52 P
Provisioning	Y	Y	Y	Y	Y	Y	Y	Y	NT	Y	Y	Y	Y	Y	Y	Y
H-A	Y	Y	Y	Y	Y	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y
Directory XIVO	N	Y	N	N	Y	N	N	N	Y	Y	Y	Y	Y	N	Y	Y
Funckeys	0	0	2	2	2	3	13	16	3	16	3	15	15	27	27	0
Supported programmable keys																
User with supervision function	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Group	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Queue	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Conference Room with supervision function	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
General Functions																
Online call recording	N	N	N	N	N	N	N	N	NYT	N	N	N	N	N	N	N
Phone status	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Sound recording	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Call recording	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Incoming call filtering	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Do not disturb	N	N	Y	SK	SK	SK	SK	SK	NYT	SK	SK	SK	SK	SK	SK	N
Group interception	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Listen to online calls	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Directory access	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Filtering Boss - Secretary	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Transfers Functions																
Blind transfer	SK	SK	HK	HK	HK	HK	HK	HK	NYT	HK	SK	SK	SK	HK	HK	SK
Indirect transfer	SK	SK	HK	HK	HK	HK	HK	HK	NYT	HK	SK	SK	SK	HK	HK	SK
Forwards Functions																
Disable all forwarding	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Enable/Disable forwarding on no answer	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Enable/Disable forwarding on busy	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Enable/Disable forwarding unconditional	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Voicemail Functions																
Enable voicemail with supervision function	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Reach the voicemail	N	N	HK	HK	HK	HK	HK	HK	NYT	HK	HK	HK	HK	HK	HK	HK
Delete messages from voicemail	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Agent Functions																
Connect/Disconnect a static agent	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Connect a static agent	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Disconnect a static agent	N	N	Y	Y	Y	Y	Y	Y	NYT	Y	Y	Y	Y	Y	Y	N
Parking Functions																
Parking	N	N	Y	Y	Y	Y	Y	Y	NYT	N	Y	Y	Y	Y	Y	N
Parking position	N	N	Y	Y	Y	Y	Y	Y	NYT	N	Y	Y	Y	Y	Y	N
Paging Functions																
Paging	N	N	Y	Y	Y	Y	Y	Y	NYT	N	Y	Y	Y	Y	Y	N

En ce qui concerne le W52P (DECT), il existe un firmware pour la base et le combiné. La base et le combiné vont probablement fonctionner s'ils n'utilisent pas la même version de firmware, bien que cela ne semble pas être officiellement recommandé. Par défaut, une station de base essaiera de mettre à niveau le firmware d'un combiné hertzien (OTA) si les conditions suivantes sont remplies :

- Combiné avec firmware 26.40.0.0.15 ou ultérieur
- Station de base avec firmware 25.40.0.0.15 ou ultérieur
- Combiné avec matériel 26.0.0.0.6 ou ultérieur

Dans le cas contraire, vous devrez mettre à jour manuellement le firmware du combiné via USB.

Dans tous les cas, vous devriez consulter la documentation de Yealink sur la mise à niveau du firmware du combiné W52x.

### Note

Certaines touches de fonction sont partagées avec les touches de ligne

Modules d'extension supportés :

- Yealink® EXP38 (pour Yealink® T26P/T28P)
- Yealink® EXP39 (pour Yealink® T26P/T28P)
- Yealink® EXP40 (pour Yealink® T46G/T48G)

Model	Tested [1]	Fkeys [2]	Wazo HA [3]	Plugin
CP860	No	0	—	xivo-yealink-v72
T23P	No	3	—	xivo-yealink-v80
T23G	Yes	3	Yes	xivo-yealink-v80
T27P	Yes	21	Yes	xivo-yealink-v80
T29G	No	27	—	xivo-yealink-v80
T49G	Yes	29	Yes	xivo-yealink-v80

## Zenitel

Model	Tested [1]	Fkeys [2]	Wazo HA [3]
IP station	Yes	1	No

## 15].Débogage Astérisk

### Condition préalable :

Pour déboguer les plantages ou les blocages d'astérisk, vous avez besoin des paquets de débogage suivants sur votre Wazo :

General rule	Version exemple	Commands
XIVO < 14.18	14.12	apt-get install xivo-fai-14.12 apt-get update apt-get install gdb apt-get install -t xivo-14.12 asterisk-dbg xivo-libscpp-dbg
XIVO >= 14.18	14.18	xivo-dist xivo-14.18 apt-get update apt-get install gdb apt-get install -t xivo-14.18 asterisk-dbg xivo-libscpp-dbg xivo-dist xivo-five
XIVO >= 16.16	17.15	xivo-dist wazo-17.15 apt-get update apt-get install gdb libc6-dbg apt-get install -t wazo-17.15 asterisk-dbg xivo-libscpp-dbg xivo-dist phoenix
XIVO >= 18.13	18.13	wazo-dist wazo-18.13 apt-get update apt-get install gdb libc6-dbg apt-get install -t wazo-18.13 asterisk-dbg wazo-libscpp-dbg wazo-dist phoenix-stretch

### **Il y a donc un problème avec Asterisk. Et maintenant quoi ?**

**1].**Renseignez-vous sur l'heure de l'incident auprès des personnes les plus susceptibles d'être informées.

**2].**Déterminez s'il y a eu une erreur de segmentation

**1).**La commande `grep segfault /var/log/syslog` devrait retourner une ligne comme celle-ci :

```
Oct 16 16:12:43 xivo-1 kernel: [10295061.047120] asterisk[1255]: segfault at e ip b751aa6b sp b5ef14d4 error 4 in libc-2.11.3.so[b74ad000+140000]
```

**2).**Notez l'heure exacte de l'incident à partir de la ligne de segmentation.

**3).**Suivez la procédure de débogage d'astérisk Crash.

**3].**Si vous observez certains des symptômes courants suivants, suivez la procédure de gel de l'astérisk de débogage.

- La sortie de la commande `service asterisk status` indique que le PBX Asterisk est en cours d'exécution.
- Plus aucun appel n'est distribué et les téléphones sont acheminés vers `No Service`.
- La commande `core show channels` retourne uniquement les en-têtes (pas de données).

**4].**Récupérez les logs d'Asterisk pour le jour du crash (assurez-vous que le fichier n'était pas déjà logrotated) :

```
cp -a /var/log/asterisk/full /var/local/`date +"%Y%m%d"``-`hostname`-asterisk-full.log
```

**5].**Récupérez les logs xivo-ctid pour le jour du crash (assurez-vous que le fichier n'était pas déjà logrotated) :

```
cp -a /var/log/xivo-ctid.log /var/local/`date +"%Y%m%d"``-`hostname`-xivo-ctid.log
```

**6].**Ouvrez un nouveau problème sur le bugtracker avec les informations suivantes

- Tracker : Bug
- Statut : Nouveau
- Catégorie : Astérisik
- En versions : La version de votre installation Wazo où le crash/freeze s'est produit
- Sujet : `Asterisk Crash` ou `Asterisk Freeze`
- Description : Ajoutez autant de contexte que possible, si possible, un scénario qui mène au problème, la date et l'heure du problème, où l'on peut aller chercher les journaux et remonter le chemin inverse.
- Joindre les logs et backtrace (si disponible) au ticket (le problème doit être enregistré, puis édité et les fichiers attachés à un commentaire).
- NE PAS joindre le fichier de base publiquement ! Il peut contenir des informations sensibles comme des mots de passe et ne doit être partagé qu'avec des personnes en qui vous avez confiance.

## **Débogage d'un crash d'astérisik**

Lorsque l'astérisik plante, il laisse généralement un fichier noyau dans `/var/spool/asterisk/`. Vous pouvez créer une trace de retour à partir d'un fichier core nommé `core_file` avec :

```
gdb -batch -ex "bt full" -ex "thread apply all bt" /usr/sbin/asterisk core_file > bt-threads.txt
```

## **Debug d'un freeze Asterisk**

Vous pouvez créer une trace arrière d'un processus d'Astérisik en cours d'exécution avec :

```
gdb -batch -ex "thread apply all bt" /usr/sbin/asterisk $(pidof asterisk) > bt-threads.txt
```

Si votre version de l'Astérisik a été compilée avec le drapeau `DEBUGUG_THREADS`, vous pouvez obtenir plus d'informations sur les verrous avec :

```
asterisk -rx "core show locks" > core-show-locks.txt
```

## Note

Déboguer le gel sans cette information est généralement beaucoup plus difficile.

En option, d'autres informations qui peuvent être intéressantes :

- la sortie d'Astérisk `asterisk -rx 'core show channels'`
- le journal verbeux d'Astérisk juste avant le gel

## **Recompiler Astérisk**

Il est relativement simple de recompiler la version Astérisk de votre Wazo avec l'option `DEBUG_THREADS` et `DONT_OPTIMIZE`.

Les étapes sont les suivantes :

**1].Dé-commentez** la ligne `deb-src` pour les sources Wazo :

```
sed -i 's/^# *deb-src/deb-src/' /etc/apt/sources.list.d/xivo*
```

**2].Récupérez** le paquet source avec Astérisk :

```
mkdir -p ~/ast-rebuild
cd ~/ast-rebuild
apt-get update
apt-get install -y build-essential
apt-get source asterisk
```

**3].Installez** les dépendances de compilation :

```
apt-get build-dep -y asterisk
```

**4].Activez** les indicateurs `DEBUG_THREADS` et `DONT_OPTIMIZE` :

```
cd <asterisk-source-folder>
vim debian/rules
```

**5].Mettez à jour** le journal des modifications en ajoutant `+debug1` dans la version du paquet :

```
vim debian/changelog
```

**6].Recompilez** les paquets binaires avec Astérisk :

```
dpkg-buildpackage -us -uc
```

Cela créera quelques fichiers.deb dans le répertoire parent, que vous pouvez installer via `dpkg`.

## **Recompiler une version vanille d'Asterisk (Wazo < 17.17)**

Il est parfois utile de produire une version "vanille" d'Asterisk, c'est-à-dire une version d'Asterisk qui n'a aucun des correctifs Wazo appliqués, pour s'assurer que le problème est présent dans le code original. C'est nécessaire avant d'ouvrir un ticket sur l'Asterisk issue tracker.

La procédure est similaire à celle décrite ci-dessus. Avant d'appeler `dpkg-buildpackage`, il vous suffit de faire :

**1].**Assurez-vous que `quilt` est installé :

```
apt-get install -y quilt
```

**2].**Unapply tous les correctifs actuellement appliqués :

```
quilt pop -a
```

**3].**Supprimez toutes les lignes de `debian/patches/series` :

```
truncate -s0 debian/patches/series
```

Lors de l'installation d'une version vanille d'Asterisk sur un XiVO 16.08 ou antérieur, vous devez arrêter la surveillance, sinon il redémarrera l' Asterisk toutes les quelques minutes.

## **Recompiler une version vanille d'Asterisk (Wazo >= 17.17)**

Wazo propose une version vanille d'Asterisk, compilée avec le drapeau `DONT_OPTIMIZE`. Cela rend le dépôt des rapports de bug sur Asterisk beaucoup plus facile.

Notez que cette version d'Asterisk perd certaines fonctionnalités spécifiques à Wazo. Les fonctionnalités supprimées incluent :

- Routage basé sur les compétences des files d'attente
- Consultation des messages vocaux via l'API REST
- Transferts d'appels via Wazo Client ou REST API

Pour installer la version vanille d'Asterisk (remplacer 17.17 par votre version actuelle de Wazo) :

```
xivo-dist wazo-17.17
apt-get update
apt-get install -t wazo-17.17 asterisk-vanilla asterisk-vanilla-dbg
xivo-fix-paths-rights
xivo-dist phoenix
```

Cette commande devrait remplacer le paquet `asterisk` par `asterisk-vanilla`

Une fois les paquets installés, vous pouvez reproduire le plantage et extraire les logs de backtrace du fichier core dump. Ces fichiers peuvent ensuite être utilisés pour déposer un rapport de bogue sur Asterisk.

Pour annuler cette modification, réinstallez `asterisk` (remplacez 17.17 par votre version actuelle de Wazo) :

```
xivo-dist wazo-17.17
apt-get update
apt-get install -t wazo-17.17 asterisk
xivo-fix-paths-rights
xivo-dist phoenix
```

## Exécuter Asterisk sous Valgrind

### 1]. Installez Valgrind :

```
apt-get install valgrind
```

### 2]. Recompiler l'astérisque avec l'indicateur DONT\_OPTIMIZE.

3]. Éditez `/etc/asterisk/modules.conf` pour que l'astérisque ne charge pas les modules inutiles. Cette étape est facultative. Il accélère (sensiblement) le démarrage de l'astérisque et facilite souvent l'analyse de la sortie de valgrind, puisqu'il y a moins de bruit.

4]. Éditez `/etc/asterisk/asterisk.conf` et commentez l'option `highpriority`. Cette étape est facultative.

### 5]. Arrêter la surveillance et l'astérisque :

```
monit quit  
service asterisk stop
```

6]. Arrêtez tous les services Wazo inutiles. Par exemple, il peut être utile d'arrêter xivo-ctid, pour qu'il n'interagisse pas avec l'astérisque via l'AMI.

7]. Copiez le fichier valgrind.supp dans /tmp. Le fichier valgrind.supp se trouve dans le répertoire contrib du code source de l'astérisque.

### 8]. Exécutez valgrind dans le répertoire /tmp :

```
cd /tmp  
valgrind --leak-check=full --log-file=valgrind.txt --suppressions=valgrind.supp --vgdb=no  
asterisk -G asterisk -U asterisk -fnc
```

Notez que lorsque vous terminez l'astérisque avec Control-C, l'astérisque ne décharge pas les modules avant de quitter. Cela signifie que vous pourriez avoir beaucoup d'erreurs de mémoire "peut-être perdues". Si vous savez déjà quels modules sont responsables de la fuite ou du bogue de mémoire, vous devez explicitement le décharger avant de terminer l'astérisque.

L'exécution de l'astérisque sous valgrind nécessite beaucoup de mémoire supplémentaire, assurez-vous donc d'avoir suffisamment de RAM.

### Liens externes

<https://wiki.asterisk.org/wiki/display/AST/Debugging>

<http://blog.wazo.community/visualizing-asterisk-deadlocks.html>

<https://wiki.asterisk.org/wiki/display/AST/Valgrind>

## 16].Débogage des daemons

Pour activer le mode débogage, ajoutez `debug: true` dans le fichier de configuration du daemon. La sortie sera disponible dans le fichier journal du daemon.

Il est également possible d'exécuter le daemon Wazo, en ligne de commande. Cela lui permettra de fonctionner en premier plan et en mode débogage. Pour voir comment l'utiliser, tapez :

```
xivo-{name} -h
```

Notez que c'est une bonne idée d'arrêter la surveillance avant d'exécuter un daemon en avant-plan :

```
systemctl stop monit.service
```

### xivo-confgend

```
twistd -no -u xivo-confgend -g xivo-confgend --python=/usr/bin/xivo-confgend --logger  
xivo_confgen.bin.daemon.twistd_logs
```

Pas de mode de débogage en confgend.

### Xivo-provd

```
twistd -no -u xivo-provd -g xivo-provd -r epoll --logger provd.main.twistd_logs xivo-provd -s -v
```

- -s pour la journalisation sur stderr
- -v pour verbose

### Consul

```
sudo -u consul /usr/bin/consul agent -config-dir /etc/consul/xivo -pid-file  
/var/run/consul/consul.pid
```

Consul enregistre sa sortie dans `/var/log/syslog` pour obtenir la sortie de consul utiliser uniquement consul monitor :

```
2015/08/03 09:48:25 [INFO] consul: cluster leadership acquired  
2015/08/03 09:48:25 [INFO] consul: New leader elected: this-xivo  
2015/08/03 09:48:26 [INFO] raft: Disabling EnableSingleNode (bootstrap)  
2015/08/03 11:04:08 [INFO] agent.rpc: Accepted client: 127.0.0.1:41545
```

#### Note

Le fichier ca- peut être différent si vous utilisez des certificats HTTPS personnalisés.



## **Mongooseim**

```
echo "{loglevel, 5}." >> /etc/mongooseim/wazo.cfg  
systemctl restart mongooseim
```

Fichier journal : `/var/log/mongooseim/ejabberd.log` Ce fichier ne contient presque rien tant que le niveau de journalisation n'est pas défini.

Vous ne devez pas laisser le niveau de journalisation activé dans la production trop longtemps. Pour le désactiver :

```
sed -i '/loglevel, 5/d' /etc/mongooseim/wazo.cfg  
systemctl restart mongooseim
```

## 17].Dépannage

La liste des bogues en cours se trouve sur le tracker officiel de Wazo.

### **Transferts par DTMF**

Lorsque vous transférez un appel en utilisant DTMF (\*1), vous obtenez une erreur d'extension invalide lors de la numérotation de l'extension.

La solution à ce problème consiste à créer un sous-programme de pré-traitement et à l'affecter aux destinations où vous rencontrez le problème.

Sous Services › IPBX › IPBX › Configuration IPBX › Les fichiers de configuration ajoutent un nouveau fichier contenant le dialplan suivant :

```
[allow-transfer]
exten = s,1,NoOp(## Setting transfer context ##)
same = n,Set(__TRANSFER_CONTEXT=<internal-
context>)
same = n,Return()
```

N'oubliez pas de remplacer <internal-context> par votre contexte interne.

Certains endroits où vous voudrez peut-être ajouter ce sous-programme de pré-traitement sont sur les files d'attentes et les appels sortants pour pouvoir transférer la personne appelée vers un autre poste.

### **Détection de fax**

Wazo ne prend pas actuellement en charge la détection de fax. Ce qui suit décrit une solution de contournement pour utiliser cette fonction. Le comportement est de répondre à tous les appels entrants (externes), attendre un certain nombre de secondes (4 dans cet exemple) : si un fax est détecté, le recevoir sinon acheminer l'appel normalement.

#### Note



Cette solution de contournement ne fonctionne que :

- sur les appels entrants vers un Utilisateur (et un Utilisateur uniquement),
- si le tronc entrant est un tronc DAHDI ou SIP,
- si l'utilisateur dispose d'une messagerie vocale qui est activée et dont le champ email est rempli
- Sachez que cette solution de contournement ne survivra probablement pas à une mise à niveau.

**1].** Dans l'interface Web et sous Services ▸ IPBX ▸ IPBX ▸ IPBX configuration ▸ Les fichiers de configuration ajoutent un nouveau fichier nommé fax-detection.conf contenant le dialplan suivant :

```
;; Fax Detection
[pre-user-global-faxdetection]
exten = s,1,NoOp(Answer call to be able to detect fax if call is external AND user has an email configured)
same = n,GotoIf("${XIVO_CALLORIGIN}" = "extern")?:return)
same = n,GotoIf("${XIVO_USEREMAIL}"):return)
same = n,Set(FAXOPT(faxdetect)=yes) ; Activate dynamically fax detection
same = n,Answer()
same = n,Wait(4) ; You can change the number of seconds it will wait for fax (4 to 6 is good)
same = n,Set(FAXOPT(faxdetect)=no) ; If no fax was detected deactivate dynamically fax detection (needed if you want directmedia to work)
same = n(return),Return()

exten = fax,1,NoOp(Fax detected from ${CALLERID(num)} towards ${XIVO_DSTNUM} - will be sent upon reception to ${XIVO_USEREMAIL})
same = n,GotoIf("${CHANNEL(channeltype)}" = "DAHDI")?changeechocan:continue)
same = n(changeechocan),Set(CHANNEL(echocan_mode)=fax) ; if chan type is dahdi set echo canceller in fax mode
same = n(continue),Gosub(faxtomail,s,1(${XIVO_USEREMAIL}))
```

**2].** Dans le fichier `/etc/xivo/asterisk/xivo_globals.conf` définissez la sous-routine utilisateur globale sur `pre-user-global-faxdetection` : cette sous-routine sera exécutée chaque fois qu'un utilisateur est appelé :

```
XIVO_PRESUBR_GLOBAL_USER = pre-user-global-faxdetection
```

**3].** Recharger la configuration de l'astérisque (pour dialplan et dahdi) :

```
asterisk -rx 'core reload'
```

## **Le serveur CTI se termine de manière inattendue**

Si vous constatez que votre serveur CTI se termine parfois de manière inattendue avec le message suivant dans `/var/log/xivo-ctid.log` :

```
(WARNING) (main): AMI: CLOSING
```

Dans ce cas, il se peut qu'un astérisque génère beaucoup de données en peu de temps sur l'AMI alors que le serveur CTI est occupé à traiter autre chose et n'est pas en train de lire activement depuis sa connexion AMI.

Si le serveur CTI prend trop de temps avant de consommer certaines données de la connexion AMI, l'astérisque fermera la connexion AMI. Le serveur CTI se ferme automatiquement lorsqu'il détecte la perte de la connexion à l'AMI.

Il y a une solution à ce problème appelé ami-proxy, qui est un processus qui tamponne la connexion AMI entre le serveur CTI et l'astérisque. Ceci ne devrait être utilisé qu'en dernier recours, car cela augmente la latence entre les processus et ne résout pas le problème fondamental.

Pour activer l'ami-proxy, vous devez :

**1].** Ajoutez un fichier `/etc/systemd/system/xivo-ctid.service.d/ami-proxy.conf` :

```
mkdir -p /etc/systemd/system/xivo-ctid.service.d
cat >/etc/systemd/system/xivo-ctid.service.d/ami-proxy.conf <<EOF
[Service]
Environment=XIVO_CTID_AMI_PROXY=1
EOF
systemctl daemon-reload
```

**2].** Redémarrez le serveur CTI :

```
systemctl restart xivo-ctid.service
```

Si vous êtes sur un cluster Wazo, vous devez faire la même procédure sur l'esclave si vous voulez que l'ami-proxy soit également activé sur l'esclave.

Pour désactiver l'ami-proxy :

```
rm /etc/systemd/system/xivo-ctid.service.d/ami-proxy.conf
systemctl daemon-reload
systemctl restart xivo-ctid.service
```

## **Agents recevant deux appels ACD**

Un agent peut parfois recevoir plus d'un appel ACD en même temps, même si les files d'attente dans lesquelles il se trouve ont le paramètre "ringinuse" réglé sur no (par défaut).

Ce comportement est causé par un bug dans l'astérisque :

<https://issues.asterisk.org/jira/browse/ASTERISK-16115>

Il est possible de contourner ce bogue dans Wazo en ajoutant une sous-routine d'agent. La sous-routine peut être définie globalement ou par agent :

```
[pre-limit-agentcallback]
exten = s,1,NoOp()
same = n,Set(LOCKED=${LOCK(agentcallback-${XIVO_AGENT_ID})})
same = n,GotoIf(${LOCKED}?not-locked,1)
same = n,Set(GROUP(agentcallback)=${XIVO_AGENT_ID})
same = n,Set(COUNT=${GROUP_COUNT(${XIVO_AGENT_ID}@agentcallback)})
same = n,NoOp(${UNLOCK(agentcallback-${XIVO_AGENT_ID})})
same = n,GotoIf($[ ${COUNT} <= 1 ]?too-many-calls,1)
same = n,Return()

exten = not-locked,1,NoOp()
same = n,Log(ERROR,Could not obtain lock)
same = n,Wait(0.5)
same = n,Hangup()

exten = too-many-calls,1,NoOp()
same = n,Log(WARNING,Not calling agent ID/${XIVO_AGENT_ID} because already in use)
same = n,Wait(0.5)
same = n,Hangup()
```

Cette solution de contournement ne s'applique qu'aux files d'attente avec les membres agents ; elle ne fonctionne pas pour les files d'attente avec les membres utilisateurs.

De plus, la sous-routine empêche l'astérisque d'appeler un agent deux fois en raccrochant le deuxième appel. Dans les statistiques de l'agent, il sera affiché comme un appel non répondu par l'agent.

## **Erreurs de localisation PostgreSQL**

La base de données et le cluster de base de données sous-jacent utilisé par Wazo sont sensibles à la configuration des paramètres linguistiques du système. La locale utilisée par la base de données et le cluster de base de données est définie lorsque Wazo est installé. Si vous changez la locale de votre système sans porter une attention particulière à PostgreSQL, vous pourriez rendre la base de données et le cluster de base de données temporairement inutilisables.

Lorsque vous travaillez avec locale et PostgreSQL, il y a quelques petites choses utiles à savoir :

- `locale -a` pour voir la liste des locales actuellement disponibles sur votre système
- `locale` pour afficher des informations sur la locale actuelle de votre shell
- `grep ^lc_ /etc/postgresql/9.6/main/postgresql.conf` pour voir la configuration locale de votre cluster de base de données
- `sudo -u postgres psql -l` pour voir la locale de vos bases de données
- le fichier `/etc/locale.gen` et la commande `locale-gen` associée pour configurer les paramètres locaux du système disponibles
- `systemctl restart postgresql.service` pour redémarrer votre cluster de base de données
- le fichier journal PostgreSQL situé dans `/var/log/postgresql/postgresql-9.6-main.log`

### **Note**

Vous pouvez utiliser n'importe quelle locale avec Wazo tant qu'elle utilise un encodage UTF-8.

## **Le cluster de base de données ne démarre pas**

Si le cluster de base de données ne démarre pas et que vous avez les erreurs suivantes dans votre fichier journal :

```
LOG:  invalid value for parameter "lc_messages": "en_US.UTF-8"
LOG:  invalid value for parameter "lc_monetary": "en_US.UTF-8"
LOG:  invalid value for parameter "lc_numeric": "en_US.UTF-8"
LOG:  invalid value for parameter "lc_time": "en_US.UTF-8"
FATAL:  configuration file "/etc/postgresql/9.6/main/postgresql.conf" contains errors
```

Cela signifie que la locale configurée dans `postgresql.conf` (ici `en_US.UTF-8`) n'est pas disponible sur votre système, c'est-à-dire qu'elle n'affiche pas la sortie `locale -a`. Vous avez deux choix pour résoudre ce problème :

- soit rendre la locale disponible en la dé-commentant dans le fichier `/etc/locale.gen` et en exécutant `locale-gen`
- ou modifiez le fichier `/etc/postgresql/9.6/main/postgresql.conf` pour définir les différentes options `lc_*` à une locale disponible sur votre système

Une fois fait, redémarrez votre cluster de base de données.

## **Impossible de se connecter à la base de données**

Si le cluster de base de données est activé mais que vous obtenez l'erreur suivante lorsque vous essayez de vous connecter à la base de données `asterisk` :

```
FATAL: database locale is incompatible with operating system
DETAIL: The database was initialized with LC_COLLATE "en_US.UTF-8", which is not recognized by
setlocale().
HINT: Recreate the database with another locale or install the missing locale.
```

Cela signifie généralement que la locale de la base de données n'est pas disponible actuellement sur votre système. Vous avez deux choix pour résoudre ce problème :

- soit rendre la locale disponible en la dé-commentant dans le fichier `/etc/locale.gen` , en exécutant `locale-gen` et en redémarrant votre cluster de base de données
- ou recréer la base de données en utilisant une autre locale

## **Erreur lors de la mise à niveau**

Dans ce cas, vous êtes le plus souvent dans l'un des cas décrits ci-dessus. Vérifiez votre fichier journal.

## **Erreur lors de la restauration d'une sauvegarde de base de données**

Si lors d'une restauration de base de données, vous obtenez l'erreur suivante :

```
pg_restore: [archiver (db)] Error while PROCESSING TOC:
pg_restore: [archiver (db)] Error from TOC entry 4203; 1262 24745 DATABASE asterisk asterisk
pg_restore: [archiver (db)] could not execute query: ERROR: invalid locale name: "en_US.UTF-8"
    Command was: CREATE DATABASE asterisk WITH TEMPLATE = template0 ENCODING = 'UTF8' LC_COLLATE
= 'en_US.UTF-8' LC_CTYPE = 'en_US.UTF-8';
```

Cela signifie généralement que la sauvegarde de votre base de données a une locale qui n'est actuellement pas disponible sur votre système. Vous avez deux choix pour résoudre ce problème :

- soit rendre la locale disponible en la dé-commentant dans le fichier `/etc/locale.gen` , en exécutant `locale-gen` et en redémarrant votre cluster de base de données
- ou si vous voulez restaurer votre sauvegarde en utilisant une autre locale (par exemple `fr_FR.UTF-8` ), alors restaurez votre sauvegarde en utilisant les commandes suivantes :

```
sudo -u postgres dropdb asterisk
sudo -u postgres createdb -l fr_FR.UTF-8 -O asterisk -T template0 asterisk
sudo -u postgres pg_restore -d asterisk asterisk-*.dump
```

## **Erreur lors de la réplication maître-esclave**

Alors la base de données esclave n'utilise probablement pas un encodage UTF-8. Vous devrez recréer la base de données en utilisant une autre locale

## **Modification de la locale (LC\_COLLATE et LC\_CTYPE) de la base de données**

Si vous avez décidé de changer la locale de votre base de données, vous devez :

- assurez-vous d'avoir assez d'espace sur votre disque dur, plus précisément dans le système de fichiers contenant le répertoire `/var/lib/postgresql` . Vous aurez, pour un instant, deux copies de la base de données avec `asterisk` .
- se préparer à une interruption de service. La procédure exige que les services soient redémarrés deux fois, et les performances du système seront dégradées pendant la création de la base de données avec la nouvelle locale, ce qui peut prendre quelques heures si vous avez une très grande base de données.
- assurez-vous que la nouvelle locale est disponible sur votre système, c'est-à-dire qu'elle apparaît dans la sortie de `locale -a`

Utilisez ensuite les commandes suivantes (en remplaçant `fr_FR.UTF-8` par votre locale) :

```
wazo-service restart all
sudo -u postgres createdb -l fr_FR.UTF-8 -O asterisk -T template0 asterisk_newlocale
sudo -u postgres pg_dump asterisk | sudo -u postgres psql -d asterisk_newlocale
wazo-service stop
sudo -u postgres psql <<'EOF'
DROP DATABASE asterisk;
ALTER DATABASE asterisk_newlocale RENAME TO asterisk;
EOF
wazo-service start
```

Vous devez également modifier le fichier `/etc/postgresql/9.6/main/postgresql.conf` pour définir les différentes options `lc_*` à la nouvelle valeur locale.

Pour plus d'informations, consultez la documentation officielle sur le support de la localisation PostgreSQL. (<https://www.postgresql.org/docs/9.6/charset.html>)

## **Lancer un appel depuis la console Asterisk**

Il est parfois utile de faire sonner un téléphone à partir de la console avec astérisque. Par exemple, si vous voulez appeler l'extension `1234` en contexte par `default` :

```
channel originate Local/1234@default extension 42@xivo-callme
```

## Capture de paquets réseau

Dans certains cas extrêmes, la capture de paquets peut être très utile pour savoir ce qui se passe entre Wazo et d'autres équipements (téléphones, trunk, etc.).

### #Capture locale, pour analyse ultérieure :

```
# change interface eth0 and filter 'udp port 5060' as you wish
tcpdump -i eth0 -w /tmp/wazo.pcap udp port 5060
```

### #Capture de paquets à distance, transmis à Wireshark via SSH :

```
# install dumpcap on the server wazo.example.com
ssh wazo.example.com apt-get install -y wireshark-common

# run the capture on interface eth0, for SIP packets only (UDP port 5060)
wireshark -k -i <(ssh wazo.example.com "dumpcap -P -i eth0 -w - -f 'udp port 5060'")
```

## Obtenir de l'aide

Parfois, il n'est tout simplement pas possible de résoudre un problème par vous-même. Dans ce cas, vous aurez très probablement besoin de l'aide d'une personne extérieure à votre réseau.

*ngrok* peut être utilisé pour donner accès à quelqu'un en dehors de votre réseau à votre serveur Wazo.

Pour rendre cela possible, vous devrez suivre ces 4 étapes faciles.

- Créer un compte sur ngrok
- Installez ngrok sur votre serveur Wazo :

### Sur un serveur 32 bits :

```
wget https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-386.zip
unzip ngrok-stable-linux-386.zip
```

### Sur un serveur 64 bits :

```
wget https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip
unzip ngrok-stable-linux-amd64.zip
```

- Ajoutez votre jeton ngrok (donné lors de votre inscription sur ngrok)

```
./ngrok authtoken <YOUR AUTH TOKEN>
```



- Ajoutez les accès SSH et HTTPS dans votre configuration ngrok

```
cat << EOF >> ~/.ngrok2/ngrok.yml
tunnels:
  webi:
    addr: 443
    proto: tcp
  ssh:
    addr: 22
    proto: tcp
EOF
```

- Démarrer ngrok

```
./ngrok start --all
```

L'URL publique et les ports qui sont maintenant disponibles pour accéder à votre serveur seront affichés. Par exemple :

```
tcp://0.tcp.ngrok.io:12345 -> localhost:22
tcp://0.tcp.ngrok.io:9876 -> localhost:443
```

moyens :

n'importe qui peut utiliser cette commande SSH dans votre machine :

```
ssh root@0.tcp.ngrok.io -p 12345
```

n'importe qui peut accéder à l'interface web via : <https://0.tcp.ngrok.io:12346>

Pour arrêter ngrok, appuyez sur Ctrl-C.

### Note



Le tunnel ngrok ne survivra pas à un redémarrage du serveur, vous devrez le réinitialiser après le redémarrage.

### Avertissement



Cette configuration est un scénario typique d'une attaque **man on the middle**. Si vous ne faites pas confiance aux serveurs Ngrok, vous devriez vous en assurer :

-le certificat HTTPS est le bon, c'est-à-dire qu'il a la même empreinte digitale :

-sur le serveur : 

```
openssl x509 -text -noout -in /usr/share/xivo-certs/server.crt -sha256 -fingerprint | grep Fingerprint
```

-dans le navigateur, vérifiez les détails du certificat pour voir l'empreinte digitale

-l'empreinte de la clé SSH du serveur est correcte, lorsque SSH vous le demande lors de la première connexion (TOFU)

## **Collecte des logs**

Lors du dépannage d'un problème, il se peut que vous ayez besoin d'envoyer des journaux pour analyse.

`wazo-debug-collect` simplifie la collecte des logs :

```
apt-get update
apt-get install wazo-debug
wazo-debug-collect -o /tmp/logs.tar.gz
```

`wazo-debug-collect` rassemblera tous les logs des différents démons Wazo, y compris Asterisk, et les regroupera dans une archive. Vous pouvez ensuite envoyer cette archive pour analyse.























### **Avertissement**

Soyez prudent avant d'envoyer les logs dans un lieu public : ils peuvent contenir des informations sensibles, qui peuvent être utilisées pour se connecter à votre Wazo.

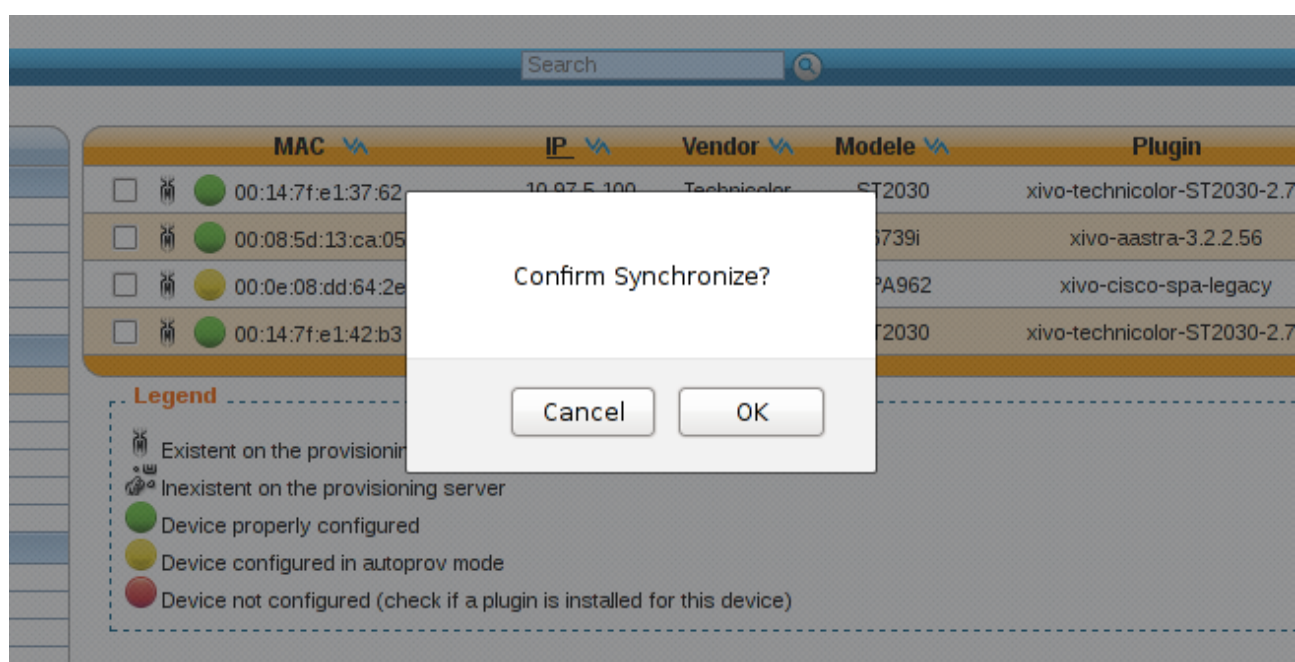
## 18].Synchroniser un appareil

Vous devez d'abord afficher la liste des périphériques.

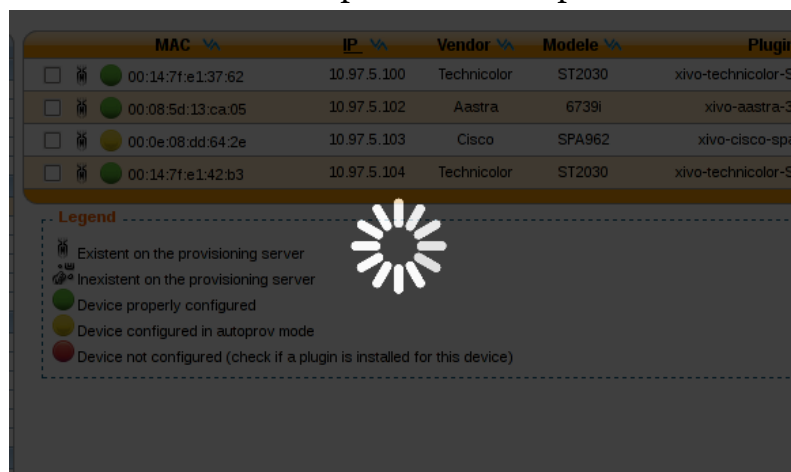
	MAC	IP	Vendor	Modele	Plugin	Action
<input type="checkbox"/>	 00:14:7f:e1:37:62	10.97.5.100	Technicolor	ST2030	xivo-technicolor-ST2030-2.74	   
<input type="checkbox"/>	 00:08:5d:13:ca:05	10.97.5.102	Aastra	6739i	xivo-aastra-3.2.2.56	   
<input type="checkbox"/>	 00:0e:08:dd:64:2e	10.97.5.103	Cisco	SPA962	xivo-cisco-spa-legacy	   
<input type="checkbox"/>	 00:14:7f:e1:42:b3	10.97.5.104	Technicolor	ST2030	xivo-technicolor-ST2030-2.74	   

Cliquez sur le bouton de synchronisation d'un appareil.

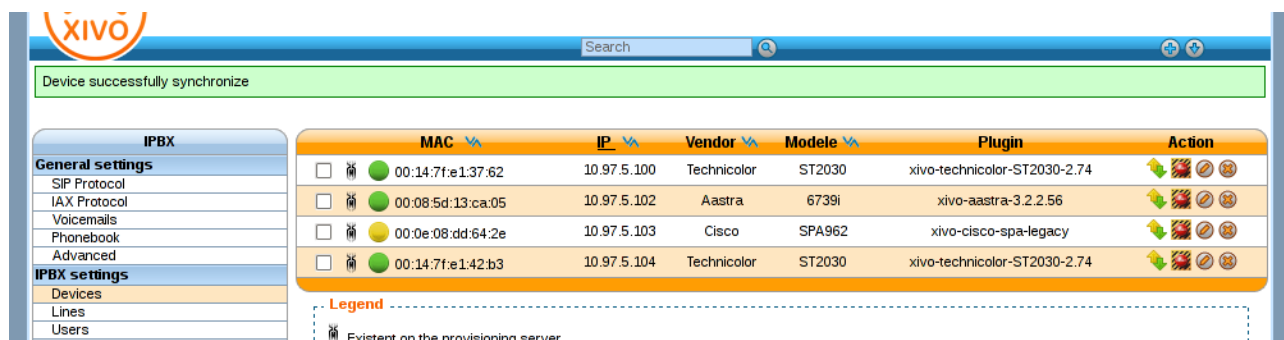
Vous verrez une fenêtre contextuelle pour confirmer la synchronisation.  
Cliquez sur le bouton <ok>.



Vous devez attendre que le processus de synchronisation complet soit terminé pour déterminer l'état retourné par l'appareil. Cela peut prendre plusieurs secondes. Il est important d'attendre et de ne rien faire pendant ce temps.



Si la synchronisation est réussie, un ballon d'information vert vous avertit de la réussite.



Si la synchronisation échoue, un ballon d'information rouge vous avertit de l'échec.

## Synchroniser plusieurs appareils

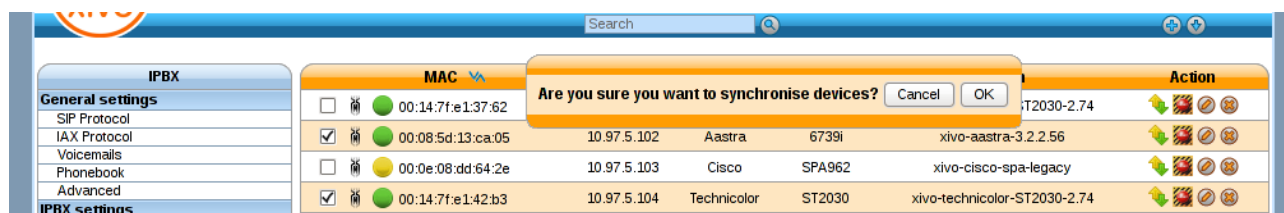
### Avertissement

Lorsque vous utilisez la synchronisation multiple, les états de retour individuels ne seront pas affichés.

Sélectionnez les appareils à synchroniser en cochant les cases correspondantes.



Une fenêtre contextuelle apparaîtra pour demander confirmation.



Si la synchronisation de masse a été envoyée avec succès aux appareils, un ballon d'information vert vous informe du succès.



## **19].Liens externe**

### **Configuration d'un environnement NAT :**

[http://documentation.wazo.community/en/latest/contributors/provisioning/nat\\_environment.html](http://documentation.wazo.community/en/latest/contributors/provisioning/nat_environment.html)

### **Plugins : c'est quoi ? | création | règles.**

<http://documentation.wazo.community/en/latest/contributors/plugins/plugins.html>

### **Générer votre propre prompt :**

[http://documentation.wazo.community/en/latest/contributors/generate\\_custom\\_prompts.html](http://documentation.wazo.community/en/latest/contributors/generate_custom_prompts.html)

### **documentation changelog :**

<http://documentation.wazo.community/en/latest/changelog.html>

### **Munin's architecture :**

<https://munin.readthedocs.io/en/latest/architecture/index.html>

### **Mise-à-jour :**

<http://documentation.wazo.community/en/latest/upgrade/upgrade.html>

### **Admin UI : plugins | installing | upgrade | remove | writing a plugin.**

[http://documentation.wazo.community/en/latest/admin\\_ui/admin\\_ui.html](http://documentation.wazo.community/en/latest/admin_ui/admin_ui.html)

### **Devices supported :**

<http://documentation.wazo.community/en/latest/ecosystem/devices.html>

### **Contact CENTER**

#### **Agent :**

[http://documentation.wazo.community/en/latest/contact\\_center/agents/agents.html](http://documentation.wazo.community/en/latest/contact_center/agents/agents.html)

#### **Queues :**

[http://documentation.wazo.community/en/latest/contact\\_center/queues/queues.html](http://documentation.wazo.community/en/latest/contact_center/queues/queues.html)

#### **Supervision :**

[http://documentation.wazo.community/en/latest/contact\\_center/supervision/supervision.html](http://documentation.wazo.community/en/latest/contact_center/supervision/supervision.html)

#### **Agent status dashboard :**

[http://documentation.wazo.community/en/latest/contact\\_center/supervision/dashboard.html](http://documentation.wazo.community/en/latest/contact_center/supervision/dashboard.html)

#### **skills-based Routing :**

[http://documentation.wazo.community/en/latest/contact\\_center/skillbasedrouting/skillbasedrouting.html](http://documentation.wazo.community/en/latest/contact_center/skillbasedrouting/skillbasedrouting.html)

**statistics :**

[http://documentation.wazo.community/en/latest/contact\\_center/statistics/statistics.html](http://documentation.wazo.community/en/latest/contact_center/statistics/statistics.html)

**Reporting :**

[http://documentation.wazo.community/en/latest/contact\\_center/reporting/reporting.html](http://documentation.wazo.community/en/latest/contact_center/reporting/reporting.html)

**Contact and Presence sharing :**

[http://documentation.wazo.community/en/latest/scalability\\_and\\_distributed\\_systems/contact\\_and\\_presence\\_sharing.html](http://documentation.wazo.community/en/latest/scalability_and_distributed_systems/contact_and_presence_sharing.html)

**phonebook sharing :**

[http://documentation.wazo.community/en/latest/scalability\\_and\\_distributed\\_systems/phonebook\\_sharing.html](http://documentation.wazo.community/en/latest/scalability_and_distributed_systems/phonebook_sharing.html)

**Message bus :**

[http://documentation.wazo.community/en/latest/api\\_sdk/message\\_bus.html](http://documentation.wazo.community/en/latest/api_sdk/message_bus.html)

**REST API :**

[http://documentation.wazo.community/en/latest/api\\_sdk/rest\\_api/rest\\_api.html](http://documentation.wazo.community/en/latest/api_sdk/rest_api/rest_api.html)

**Subroutine :**

[http://documentation.wazo.community/en/latest/api\\_sdk/subroutine.html](http://documentation.wazo.community/en/latest/api_sdk/subroutine.html)

**Websocket Service :**

[http://documentation.wazo.community/en/latest/api\\_sdk/websocket.html](http://documentation.wazo.community/en/latest/api_sdk/websocket.html)

**Community documentation :**

<http://documentation.wazo.community/en/latest/community/community.html>

**Xivo pour les nuls :**

<https://xivopourlesnuls.wordpress.com/>

**Youtube chaîne XIVO :**

<https://www.youtube.com/watch?v=EmY2KhCn418>

**Xivo configuration basiques :**

<http://www.networklab.fr/xivo-configuration-basique/>

**XIVO auto-hébergement téléphonie, pfsense, OVH.**

<http://blog.labcellar.com/2015/01/28/xivo-auto-hebergement-telephonie-pfsense-ovh/>