Séminaire

22-23 Novembre 2006

Sphynx

Historique (1)

- Sphynx 1.0
  - Sortie debut 2003
  - Même base que l'Amon-1.5
  - Utilisation freeswan-1.99
  - Générateur de configuration vpn en XML
  - Conforme au cahier des charges AGRIATES
  - possibilité de ne monter qu'un seul tunnel avec un Amon

Historique (2)

- Sphynx-1.1
  - Sortie debut 2004 en beta.
  - Basé sur une Mdk 9.1.
  - Au départ, pas d'évolution fonctionnelles puis,
    - Intégration de la gestion multi-tunnels vers Amon.
    - Passage à super-freeswan.
    - Prise en compte du nat traversal.
    - gestion des adresses privées entre un Amon et un routeur.
  - Ajout des produits permettant de gérer la haute disponibilité.
  - Ajout d'outils permettant de gérer le routage dynamique.
  - Évolutions noyau.

Bilan actuel

- Une cinquantaine de Sphynx déployés.
- Produit vieillissant :
  - problème de renouvellement des tunnels bloqués.
  - Lenteurs lors de la génération des fichiers de configuration (lié au nombre de tunnels).
  - Problèmes de stabilité ?
  - Plus de possibilités d'évolution.
  - Plus de mise à jour de sécurité.

Sphynx NG(1)

- Version 1.98 livrée en juin 2006 en version Alpha
  - Basée sur une distribution Sourcemage
  - Passage sur un noyau 2.6
    - Couche ipsec intégrée au Noyau.
  - Abandon de Freeswan au profit de Strongswan.
  - Nouveau système de templates Eole.
  - Niveau fonctionnel quasi identique.
- Actuellement, finalisation de la migration sur Ubuntu.

Sphynx NG(2)

- Strongswan (1)
  - Projet basé sur les dernières versions de freeswan
  - Intègre la gestion des certificats X509 v3
    - Support des tokens, smartCards ...
    - Meilleure prise en compte des crls.
    - Récupération et cache local des crls via http et ldap.
    - Support complet du protocole ocsp (Online Certificate Status Protocol).
    - Récupération automatique des crls (si champ URI présent dans le certificat).

Sphynx NG(3)

- Strongswan (2)
  - Gestion de la politique de sécurité ipsec étendue pouvant-être basée sur :
    - des CA intermédiaires et des wildcards.
    - des groupes avec des attributs de certificats.
    - des ports ou des groupes de ports.
    - des ip ou des plages d'ip.
  - Insertion et suppression automatiques des règles de firewall.
  - Meilleure prise en compte des algorithmes de crypto
    - aes (jusqu'à 512), blowfish, towfish, serpent ...
  - Meilleur support du nat traversal.

Sphynx NG(4)

- Strongswan (3)
  - Intégration complète du Dead Peer Detection.
  - Commande permettant le démarrage et renouvellement des connexions plus rapidement.
  - Gestion statique et dynamique (en pull ou en push) des ip virtuelles.

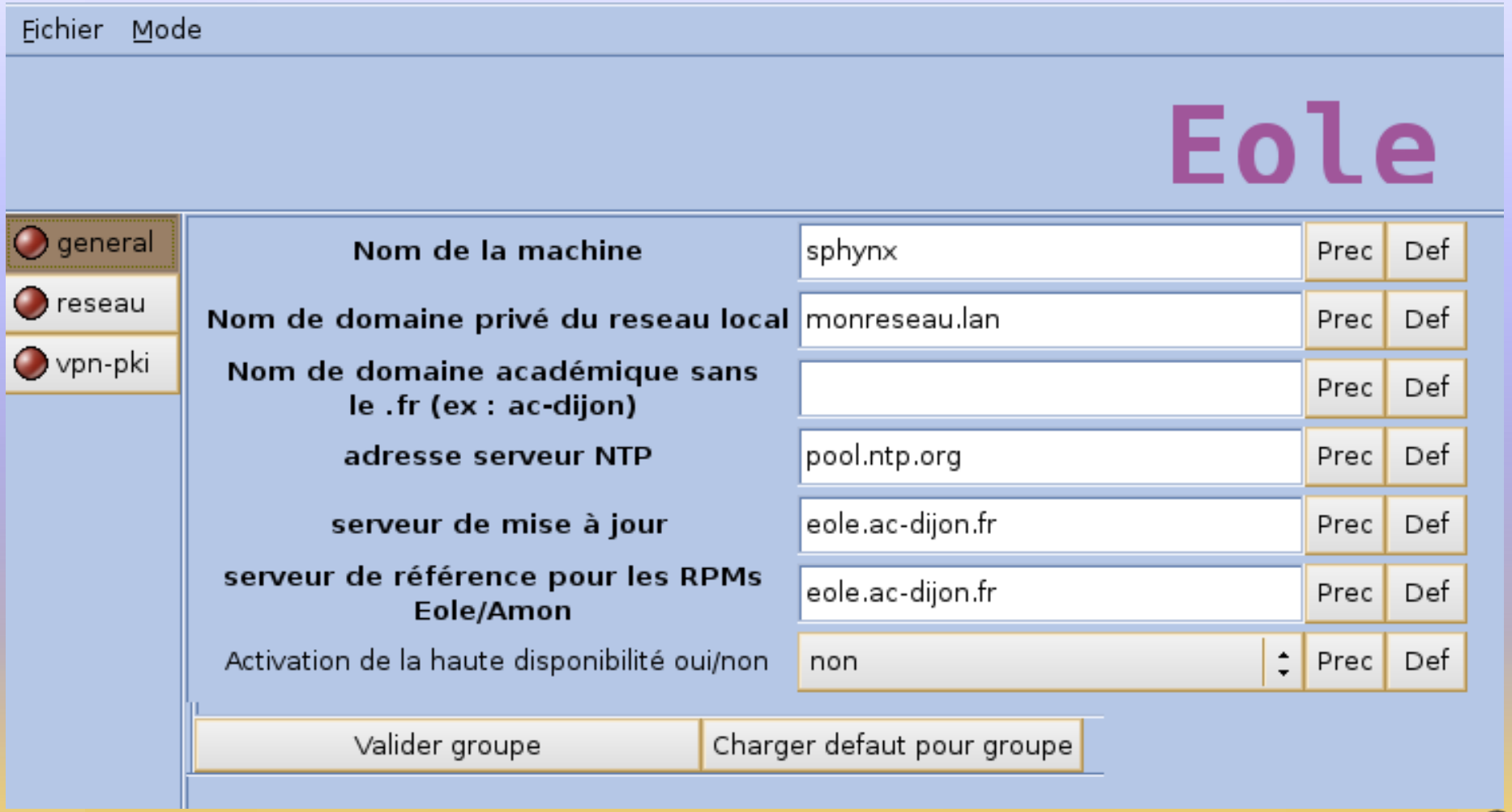
Sphynx NG(5)

- Évolutions
  - Au niveau de la configuration :
    - Intégration et gestion de la haute disponibilité (sans répartition de charge).
    - Gestion des vlans, des ip virtuels, des routes.
    - Intégration de la configuration du routage dynamique.
  - Au niveau du générateur :
    - Refonte totale de l'interface.
    - Interaction beaucoup plus forte avec Zephir.
    - Passe par une redéfinition préalable des besoins.

Sphynx NG(6)

- Autres évolutions possibles :
  - Prise en charge ipv6.
  - Prise en charge mpls.
  - Qualitée de service.
  - ...

Nouveaux dictionnaires(1)



Nouveaux dictionnaires(2)

- Fichier dictionnaire en XML

```
<?xml version="1.0" encoding="iso-8859-1"?>

<creole>
<files>
        <file name='/etc/hosts'/>
        <file name='/etc/sudoers' mode="0440"/>
<file filelist='dyn_route' name='/etc/quagga/zebra.conf'/>
<file filelist='dyn_route' name='/etc/quagga/debian.conf'
source='quagga.conf'/>
</files>
```

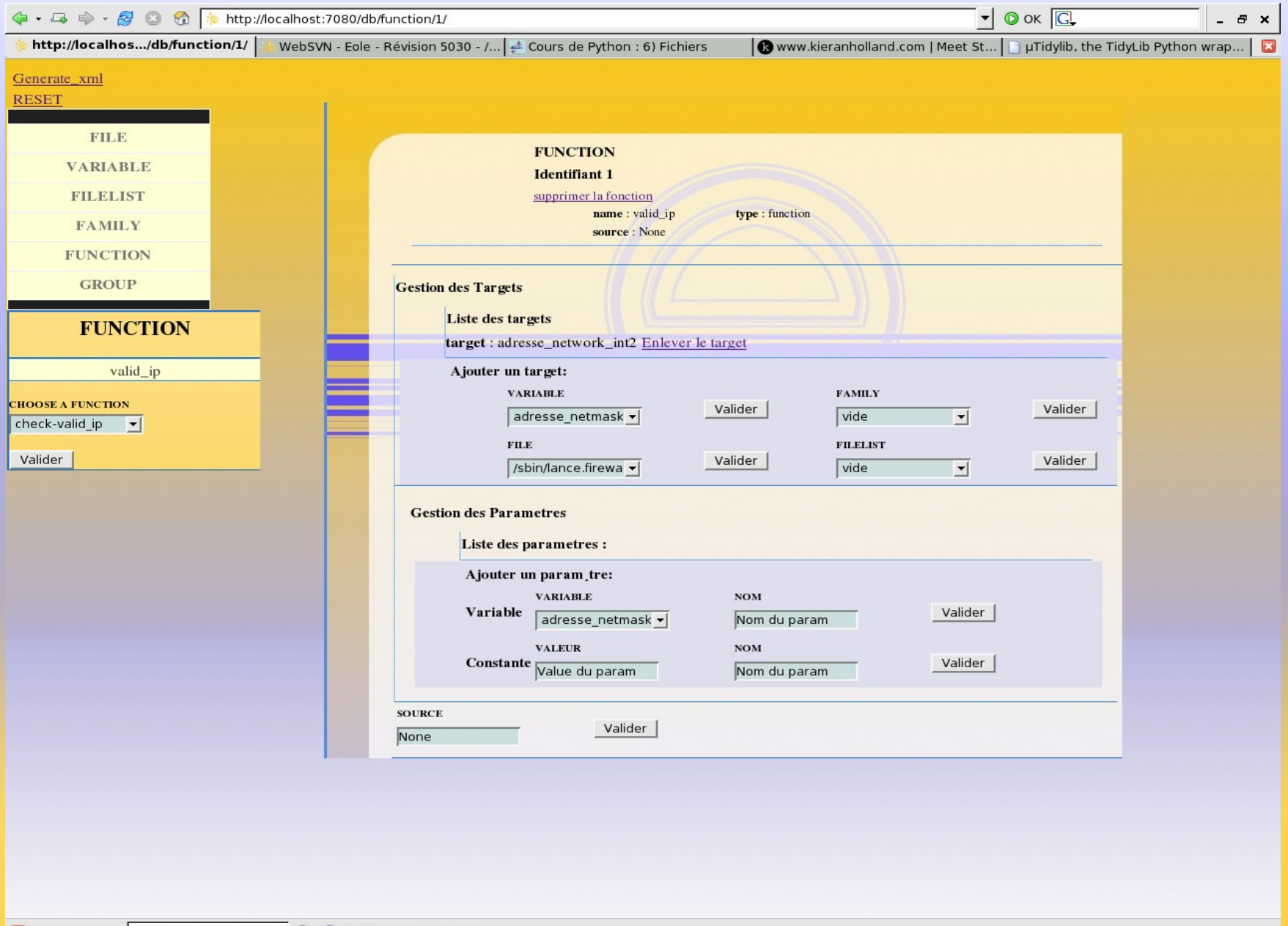
Nouveaux dictionnaires(3)

```
<variables>
    <family name='general'>
        <variable name='nom_machine' type='string'
description='Nom de la machine' >
            <value>sphynx</value>
        </variable>
        <variable name='nom_domaine_local' type='string'
description='Nom de domaine privé du reseau local' >
            <value>monreseau.lan</value>
        </variable>
```

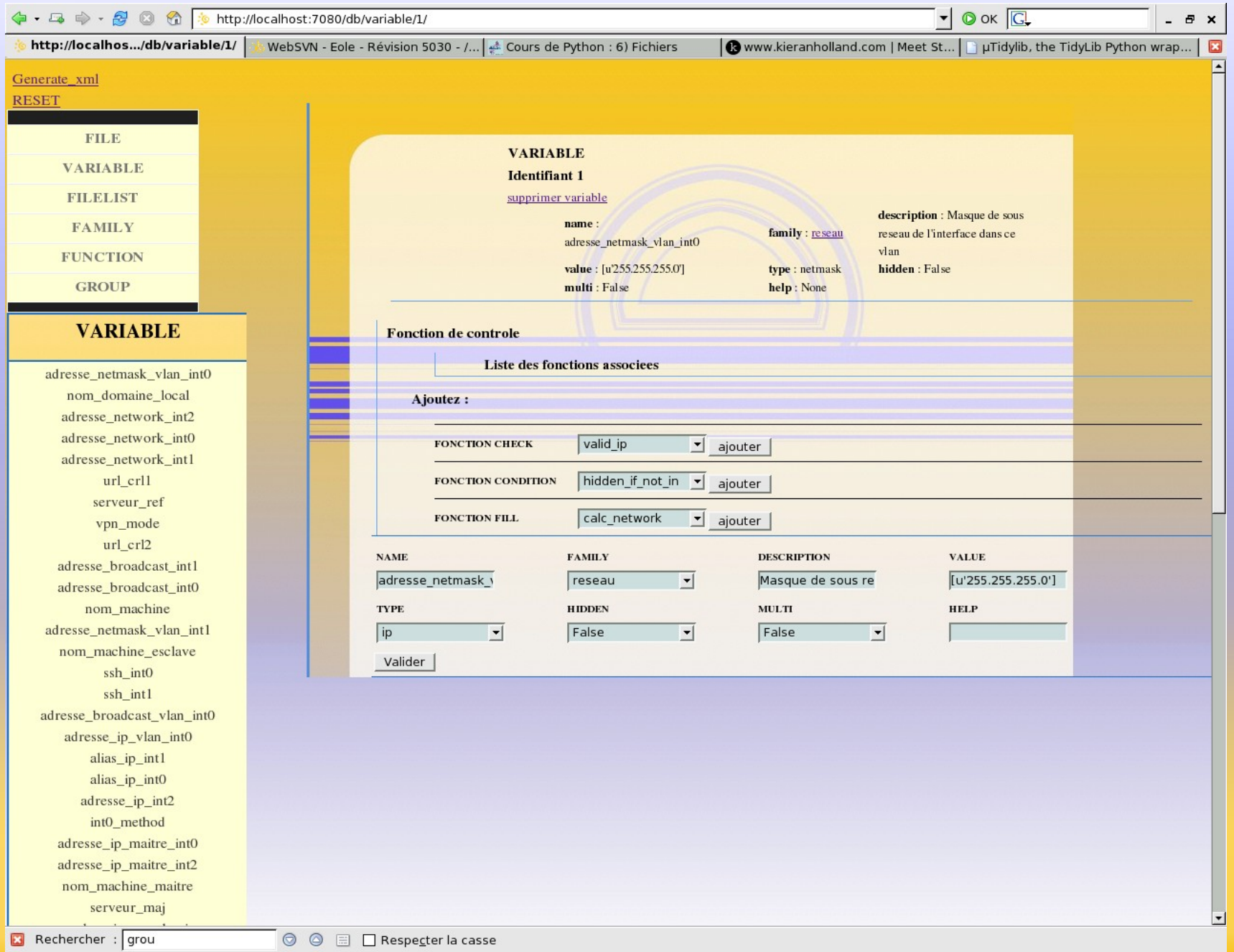
Nouveaux dictionnaires(4)

```
<constraints>
    <fill name='calc_network'
target='adresse_network_int0'>
        <param type='eole'
name='ip'>adresse_ip_int0</param>
        <param type='eole'
name='netmask'>adresse_netmask_int0</param>
    </fill>
        <check name='obligatoire' target='nom_machine'/>
        <check name="valid_enum" target='haute_dispo'>
        <param>['non','maitre','esclave']</param>
    </check>
    </constraints>
```

Nouveaux dictionnaires(5)



Nouveaux dictionnaires(6)



Nouveaux dictionnaires(7)

- Exemple d'un fichier de configuration :

```
%if %%ssh_int0 == "oui"
%for %%reseau_ssh0 in %%ip_ssh_int0
%if %%reseau_ssh0.netmask_ssh_int0 == "255.255.255.255"
sshd:%%reseau_ssh0
%else if %%reseau_ssh0.netmask_ssh_int0 != "255.255.255.255"
sshd:%%reseau_ssh0/%%reseau_ssh0.netmask_ssh_int0
%end if
%end for
%end if
```

Merci de votre attention.