

---

# EyesOfNetwork Configuration

Aide à la configuration de la solution EoN

Fernandez Sébastien



**Référence d'origine** : Eon 2.2 Configuration

**Référence actuelle** : Eon v5.0 Configuration

**État** : Terminé

**Date dernière modification** : 22 juillet 2016

*Ce fichier vous est distribué sous licence [Creative Commons Paternité version 2.5](http://creativecommons.org/licenses/by/2.5/deed.fr).  
Pour résumer : vous êtes libre de distribuer et de modifier ce fichier pour peu que vous créditiez son ou ses  
auteur(s). La mention de la licence est facultative pour les œuvres dérivées. Texte officiel de la licence:  
<http://creativecommons.org/licenses/by/2.5/deed.fr>*

---

-- Historique des modifications --

Version	Date	Responsable	Modifications
2.2	22/10/2010	Fernandez Sébastien	Création du document
2.2	18/06/2011	Fernandez Sébastien	Corrections d'erreurs
2.2	27/06/2011	Férandin Marc	Ajout de la section « Backend »
2.2	24/10/2011	Férandin Marc	Ajout de la section « GED »
3.1	21/06/2012	Fernandez Sébastien	Modification en V3.1
4.0	22/07/2013	Fernandez Sébastien	Modification en V4.0
5.0	22/07/2016	Dylan Galmiche	Modification en V5.0

---

## Sommaire

<b>1. PREAMBULE.....</b>	<b>4</b>
1.1. POURQUOI CETTE DOCUMENTATION ? .....	4
1.2. REMERCIEMENTS.....	4
1.3. AXE DE PROGRESSION.....	4
<b>2. NOMENCLATURE.....</b>	<b>5</b>
<b>3. CONFIGURATION SNMP DE LA SOLUTION .....</b>	<b>6</b>
<b>4. POSTFIX : SECTION MESSAGERIE.....</b>	<b>10</b>
<b>5. DESACTIVER L'ACCES ROOT DIRECT EN SSH .....</b>	<b>12</b>
<b>6. MODIFIER L'APPARENCE PAR DEFAUT DE THRUK.....</b>	<b>13</b>
<b>7. MODIFIER L'ACCES WEB POUR L'UTILISATION D'OUTIL TYPE NAGSTAMON .....</b>	<b>14</b>
<b>8. CONTROLE D'ACCES WEB : ACTIVATION DES COOKIES !.....</b>	<b>16</b>
<b>9. CONFIGURATION DU RSYSLOG.CONF POUR RECEVOIR LES LOGS D'AUTRES HOSTS : .....</b>	<b>24</b>
<b>10. MISE EN PLACE DE GNOKII POUR ACTIVER LES NOTIFICATIONS PAR SMS.....</b>	<b>25</b>
<b>11. MODIFICATION DU BACKEND LIVESTATUS.....</b>	<b>27</b>
11.1. INTRODUCTION .....	27
11.2. CONFIGURATION D'UN BACKEND .....	27
11.3. DECLARATION DES BACKENDS SUR LE SERVEUR CENTRAL .....	27
11.4. DECLARATION MANUELLE .....	28
11.5. DECLARATION VIA NAGVIS.....	28
11.6. AJOUTER UN HOST OU SERVICE AVEC CE BACKEND .....	28
<b>12. CONFIGURATION DE GED POUR UNE ARCHITECTURE DISTRIBUEE : .....</b>	<b>30</b>
12.1. CONFIGURATION DU SERVEUR CENTRAL.....	30
12.2. CONFIGURATION DU SERVEUR DISTANT .....	30
12.3. REDEMARRAGE DU PROCESSUS GED .....	31
12.4. VERIFICATIONS : .....	32
12.5. VISUALISATION DANS L'INTERFACE « EONWEB », ONGLET DISPONIBILITES : .....	33
<b>13. CONCLUSION : .....</b>	<b>35</b>

---

## 1. Préambule

### 1.1. Pourquoi cette documentation ?

Cette documentation permettra aux utilisateurs de la solution de mieux appréhender les possibilités d'EyesOfNetwork.

Cependant certaines connaissances basiques en Linux/Nagios/Cacti seront nécessaires pour la bonne compréhension de la documentation.

Il est donc fortement conseiller aux néophytes de se familiariser avec un environnement Linux/Nagios... de consulter la documentation d'utilisation de EON avant d'aller plus loin et même de jeter un œil à celle créée par Anthony Leduc :

« EON - Configuration et administration d'un logiciel de supervision réseau »

### 1.2. Remerciements

Avant de rentrer dans le vif du sujet, je souhaite remercier les membres actifs de la communauté de « [www.eyesofnetwork.com](http://www.eyesofnetwork.com) » notamment Anthony Leduc dont la mise en page de sa documentation « EON - Configuration et administration d'un logiciel de supervision réseau » à été largement reprise ici ainsi que les paragraphes « backup manager » et « mise a jour de EoN ».

Mes remerciements sont également destinés à Jean-Philippe Levy, Jérémie Bernard et Michael Aubertin pour avoir su développer un outil complet, accessible et pertinent. A ceux-ci s'ajoutent de nouveaux intervenants, comme par exemple Emmanuelle Texeire, responsable entre autre de la traduction de la documentation en anglais.

### 1.3. Axe de progression

Cette documentation décrira quelques possibilités offertes par la solution ainsi que des exemples.

Il ne tient qu'à vous de compléter cette documentation afin qu'elle corresponde au besoin du plus grand nombre.

Les contributeurs n'hésiteront pas à compléter la partie « historique ».

---

## 2. Nomenclature

Une ligne de commande à saisir dans la console est représentée sous cette forme :

```
/etc/init.d/xxx
```

Une information importante et qui réclame votre attention est représentée ainsi :



*Pensez à redémarrer le service*

Une information qui peut vous être utile est représentée de cette façon :



Faire « *systemctl restart service* »

Les modifications de fichiers de configuration de forme linux sont supposées faites par la commande : « vi »  
Après chaque modification il est supposé que l'utilisateur pense à le sauvegarder en faisant la combinaison  
Esc puis :wq ou Esc puis :x (important les :).

---

### 3. Configuration SNMP de la solution

La configuration SNMP de la solution est un point important, en effet si cette partie est mal paramétrée, vous risquez de ne pas avoir de remontés Cacti, Nagios, ...

3 parties distinctes bien qu'en relations entre elles :

- le fichier /etc/snmp/snmpd.conf

```
####  
# First, map the community name "EyesOfNetwork" into a "security name"  
  
#       sec.name  source          community  
com2sec notConfigUser default      EyesOfNetwork
```

Ce fichier gère la communauté et le type de SNMP activé en local sur EoN.

Par défaut la communauté est « EyesOfNetwork » en SNMP v1 et v2c. Pour la modifier remplacer le nom de communauté par celle désirée pour être cohérent avec votre parc.



*Pensez à redémarrer le service snmpd par l'interface administration eonweb (Administration -> Généralités -> Processus) ou en ligne de commande :*

```
systemctl restart snmpd
```

- le fichier /etc/snmp/snmptrapd.conf

```
ignoreauthfailure yes  
authCommunity log,execute,net EyesOfNetwork  
traphandle default /srv/eyesofnetwork/snmptr/bin/snmptrhandler
```

Ce fichier gère la communauté SNMP des « traps » reçus par EoN depuis les équipements de votre parc. De la même manière que précédemment remplacez « EyesOfNetwork » par le nom désiré.

Bien différencier SNMP de snmptrap ! Quand vous configurez un service SNMP cela implique qu'une application extérieure (dans notre cas un script Nagios) va accéder à ce service pour lire des informations, c'est de la supervision active.

Snmpttrap lui envoie directement des informations à votre serveur suite à un évènement, on parle dans ce cas de supervision passive.



Pensez à redémarrer le service `snmptrapd` par l'interface administration `eonweb` (Administration -> Généralités -> Processus) ou en ligne de commande :

```
systemctl restart snmptrapd
```

- les variables de la solution EoN : Elles définissent la communauté utilisée par nagios par exemple. Si vous avez précédemment modifiés la communauté SNMP de EoN pour correspondre à votre parc, pensez à mettre à jour la variable `$user2$` de la configuration Nagios en allant dans la section « Administration -> Configuration Nagios » de `eonweb`.

Sélectionnez maintenant « Nagios Resources ». Vous trouverez la variable `$user2$` avec la valeur « EyesOfNetwork ». Si vous avez modifié le `snmpd.conf` mettez la même chose !

The screenshot shows the 'Eyes Of Network' web interface. The main content area is titled 'EONWEB CONFIGURATOR' and 'NAGIOS RESOURCES'. It contains a list of configuration fields for Nagios resources, labeled \$USER1 through \$USER27. Each field has a text input box. The fields are arranged in two columns. The first column contains \$USER1 through \$USER10, and the second column contains \$USER17 through \$USER27. The fields are: \$USER1: /srv/eyesofnetwork/nagios/plugins; \$USER2: EyesOfNetwork; \$USER3: compte\_service\_eon\_1; \$USER4: compte\_service\_eon\_2; \$USER5: /home/nagios/sessionfiles; \$USER6: /srv/eyesofnetwork/notifier; \$USER7: ; \$USER8: ; \$USER9: ; \$USER10: /srv/eyesofnetwork/pnp4nagios; \$USER11: ; \$USER17: ; \$USER18: nom\_domaine; \$USER19: mdp\_compte\_service\_1; \$USER20: mdp\_compte\_service\_2; \$USER21: ; \$USER22: ; \$USER23: ; \$USER24: ; \$USER25: ; \$USER26: ; \$USER27: .

Une fois la variable changée, cliquez en bas sur « Update Resource Configuration ». Allez ensuite dans la section « Administration -> Appliquer la configuration ». Faites enfin un export vers Nagios via un restart du job d'export Nagios.

Vous avez maintenant la même communauté locale EoN que celle utilisée par vos commandes Nagios !

Il est de même souhaitable de modifier aussi la communauté snmp par défaut de Cacti qui est elle aussi sur « EyesOfNetwork ». Pour cela allez dans la section « Administration -> Liens externes -> Cacti » de « eonweb ».

Dans Cacti allez dans la partie « Settings»

The screenshot shows the Cacti Settings (General) page. The left sidebar contains a navigation menu with categories like 'Create', 'Management', 'Collection Methods', 'Import/Export', 'Syslog Settings', 'Configuration', and 'Utilities'. The main content area is titled 'Cacti Settings (General)' and has several tabs: 'General', 'Paths', 'Poller', 'Graph Export', 'Visual', 'Authentication', 'Mail / DNS', 'Misc', and 'Syslog'. The 'General' tab is active, showing the following settings:

- Event Logging**
  - Log File Destination:** Logfile Only (dropdown)
  - Web Events:** What Cacti website messages should be placed in the log.
    - Web SNMP Messages
    - Web RRD Graph Syntax
    - Graph Export Messages
- Poller Specific Logging**
  - Poller Logging Level:** What level of detail do you want sent to the log file. WARNING: Leaving in any other status than NONE or LOW can exhaust your disk space rapidly. (dropdown: LOW - Statistics and Errors)
  - Poller Syslog/Eventlog Selection:** If you are using the Syslog/Eventlog, What Cacti poller messages should be placed in the Syslog/Eventlog.
    - Poller Statistics
    - Poller Warnings
    - Poller Errors
- Required Tool Versions**
  - SNMP Utility Version:** The type of SNMP you have installed. Required if you are using SNMP v2c or don't have embedded SNMP support in PHP. (dropdown: NET-SNMP 5.x)
  - RRDTool Utility Version:** The version of RRDTool that you have installed. (dropdown: RRDTool 1.3.x)
- SNMP Defaults**
  - SNMP Version:** Default SNMP version for all new hosts. (dropdown: Version 2)
  - SNMP Community:** Default SNMP read community for all new hosts. (text input: EyesOfNetwork)
  - SNMP Username (v3):** The SNMP v3 Username for polling hosts. (text input: empty)
  - SNMP Password (v3):** The SNMP v3 Password for polling hosts. (text input: empty)
  - SNMP Auth Protocol (v3):** Choose the SNMPv3 Authorization Protocol. (dropdown: MD5 (default))
  - SNMP Privacy Passphrase (v3):** Choose the SNMPv3 Privacy Passphrase. (text input: empty)
  - SNMP Privacy Protocol (v3):** Choose the SNMPv3 Privacy Protocol. (dropdown: empty)

Remplacez la valeur présente dans « SNMP Community » par celle désirée. Cliquez en bas sur « Save ».

---

## 4. Postfix : section messagerie

Vous avez noté qu'EON permet d'envoyer des notifications Nagios via mail.

En effet un serveur postfix est intégré dans ce but MAIS aucun agent de distribution de courrier aux clients n'est en place. Vous pouvez donc vous servir de ce postfix pour le configurer comme relay SMTP vers votre domaine de messagerie principal de votre entreprise.

Cet exemple est générique et ne fonctionnera pas si votre serveur principal requiert une authentification pour son SMTP.



*Le fichier principal de configuration de postfix se trouve en /etc/postfix/main.cf*

Exemple de quelques variables importantes du main.cf pour « pousser » les mails :

```
myhostname = nagios.localdomain (hostname fqdn de EON)
mydomain = localdomain (fqdn de EON moins le nom d'hote)
myorigin = $myhostname (indiquera de quel « serveur » viendra le futur mail)

relay_domains = domaine de messagerie de votre entreprise

default_transport = smtp

relayhost = ip du serveur de messagerie de l'entreprise
```

Les autres variables peuvent rester par défaut. Il faut savoir que rien ne vous empêche d'ajouter un agent de distribution comme dovecot ou cyrus, ceci n'a pas été mis en place car ce n'est pas le but premier d'EON !

A l'issue de chaque modification :

```
systemctl restart postfix
```



*Points importants :*

- N'oubliez pas de vérifier que votre serveur de messagerie entreprise accepte les connexions entrantes SMTP de votre EON !
- Si vous avez dans votre main.cf une ligne du genre :

```
alias_database = hash:/etc/aliases
```

Vous aurez la possibilité de renvoyer les mails de root vers une adresse externe.

Dans le cas où votre interconnexion messagerie fonctionne, éditez ce fichier /etc/aliases et ajoutez tout en bas :

```
root : @mail_du_compte_ou_recevoir_les_mails_root
```

---

Du coup vous aurez aussi les mails root suite aux sauvegardes « backup manager » car la tâche de sauvegarde est lancée par le gestionnaire « cron » et par défaut celui envoie les logs via mail à root.



*Après chaque modification du fichier aliases, faire les commandes suivantes :*

```
newaliases
```

```
systemctl reload postfix
```

N'hésitez pas à rechercher sur la toile de plus amples renseignements, postfix étant relativement complexe avec énormément de paramètres, à vous de voir ce dont vous avez besoin.

La possibilité existe reste à l'utiliser !

---

## 5. Désactiver l'accès root direct en SSH



*Garder une console active au cas où pour ne pas vous « couper la patte » !*

Un petit interlude afin de désactiver l'accès root en ssh, faille de sécurité en soi.

En mode console, éditez le fichier `/etc/ssh/sshd_config`

Dans ce fichier repérez la ligne contenant :

```
# PermitRootLogin yes
```

Et en lieu et place :

```
PermitRootLogin no
```

Cad : enlevez le # et mettez no au lieu de yes.

Sauvegardez/quittez.

Créez un compte de « maintenance » :

```
useradd maintenance -g wheel
```

Définissez un mot de passe :

```
passwd maintenance
```

Saisissez un mot de passe et confirmez-le.

Essayez, avant de finaliser les modifications, de vous loguer via ssh avec ce compte maintenance.

Une fois le mot de passe saisi vous devriez avoir un prompt \$ et non # comme pour root.

A ce stade pour passer root depuis ce compte maintenance, faites :

```
su -
```

Si tout fonctionne, redémarrez le service ssh (en tant que root) et à l'issue personne ne pourra se loguer via ssh en root au serveur EON.

```
systemctl restart sshd
```

---

## 6. Modifier l'apparence par défaut de thruk

Par défaut lorsque vous lancez thruk, la carte est pré positionnée (en haut à droite) au niveau vue « Map layout » : « Table layout » et au niveau « group by » sur « IP address ».

Si cela ne vous convient pas il est possible de le modifier en faisant comme suit :

En console, éditez le fichier `thruk_local.conf` :

```
vi /srv/eyesofnetwork/thruk/thruk_local.conf
```

```
# set the from address used in e-mail reports
#report_from_email = User Name <example@mail.com>
pnp_export = /srv/eyesofnetwork/thruk/plugins/plugins-available/reports2/script/pnp_export.sh
</Component>

#####
# STATUSMAP
<Component Thruk::Plugin::Statusmap>
  # you may change the default map type of the statusmap here. Valid
  # types are: 'table' and 'circle'
  default_type      = circle

  # and the statusmap default group by which has to be one of:
  # 'parent', 'address', 'domain', 'hostgroup', 'servicegroup'
  default_groupby = parent
</Component>

#####
# MINEMAP
<Component Thruk::Plugin::Minemap>
  # you may change the default minemap link here
  #default_link = /thruk/cgi-bin/minemap.cgi
</Component>
```

Dans le cas présent les parties « table layout » (noté `default_type`) et « group by » (`default_groupby`) ont été respectivement positionnée à « circle » et « parent ».

Quittez et sauvez :

```
:x!
```

Relancez apache :

```
systemctl restart httpd
```

Dorénavant thruk s'ouvrira par défaut sur la vue souhaité.

---

## 7. Modifier l'accès web pour l'utilisation d'outil type Nagstamon

La solution EoN impose de se logger obligatoirement soit par mysql soit par ldap pour atteindre un de ces sites web.

Cependant pour utiliser des outils type Nagstamon (ou le plugin Mozilla Nagios) il est nécessaire de pouvoir se connecter par une authentification classique type htaccess.

Afin de ne pas perturber la solution voici un palliatif :

Créez un répertoire genre nagstamon (pour notre exemple) comme suit :

```
mkdir /srv/eyesofnetwork/nagstamon
```

Copiez dedans les fichiers cgi de nagios :

```
cp /srv/eyesofnetwork/nagios/sbin/*.cgi /srv/eyesofnetwork/nagstamon/
```

Mettez les bons droits :

```
chown -R apache:eyesofnetwork /srv/eyesofnetwork/nagstamon  
chmod -R 775 /srv/eyesofnetwork/nagstamon
```

Créez un fichier de configuration apache comme suit :

```
vi /etc/httpd/conf.d/nagstamon.conf
```

Saisissez dedans :

```
ScriptAlias /nagstamon /srv/eyesofnetwork/nagstamon/  
<Directory /srv/eyesofnetwork/nagstamon/>  
    AuthType Basic  
    AuthName "Restricted"  
    AuthUserFile /srv/eyesofnetwork/nagstamon/.htpasswd.users  
    Require valid-user  
    Options ExecCGI  
    Order allow,deny  
</Directory>
```

Faites « :wq! » pour enregistrer et quitter.

---

Il ne reste plus qu'à mettre en place l'authentification, pour cela :

```
htpasswd -c /srv/eyesofnetwork/nagstamon/.htpasswd.users admin
```

A noter que l'on garde le compte « admin » car celui-ci est dans les contact nagios et voit donc tous les équipements.

Saisissez le mot de passe voulu.

Relancez le service apache :

```
systemctl restart httpd
```

Vous pourrez dorénavant configurer votre logiciel nagstamon pour attaquer « nagios » sur

@ipEoN/nagstamon/ avec les identifiant précédemment saisi !

Ceci n'est qu'un exemple, libre à vous de l'optimiser (pas besoin de tous les cgi) ou de procéder différemment !

---

## 8. Contrôle d'accès web : activation des cookies !

Suite à la mise à jour de Nagvis, EON offre la possibilité d'activer les cookies pour Nagios, Cacti,...

Cet aspect est très intéressant dans le cadre d'une gestion fine des usagers.

En effet il sera possible qu'un utilisateur « toto » logué à eonweb en tant que toto, soit aussi reconnu « toto » par Nagvis, contact Nagios, Cacti.

Cela permettra d'activer des restrictions d'accès par utilisateurs sur les cartes Nagvis, Nagios, Weathermap ou graphes Cacti...

Pour mettre en place cette possibilité, il faut d'abord activer les cookies Nagios et Cacti comme pour Nagvis :

### **- Prenons le cas de Cacti :**

En mode console en tant que root, éditez le fichier /etc/httpd/conf.d/cacti.conf

Dans le début du fichier là où il y a cette partie :

```
Alias /cacti /srv/eyesofnetwork/cacti
<Directory /srv/eyesofnetwork/cacti>
    AuthType Basic
    AuthName "Restricted"
    AuthFormAuthoritative On
    AuthFormMySQLSocket /var/lib/mysql/mysql.sock
    AuthFormMySQLUsername eonweb
    AuthFormMySQLPassword root66
    AuthFormMySQLDB eonweb
    AuthFormMySQLTableSID sessions,users,groupright
    AuthFormMySQLFieldUID sessions.user_id
    AuthFormMySQLTableSIDCondition "`sessions`.`session_id`=$session_id AND
`sessions`.`user_id`=$user_id AND `users`.`group_id`=$group_id AND `groupright`.`group_id`=$group_id
AND `groupright`.`tab_6`='1'"
    AuthFormPageLogin /login.php
    AuthFormSessionCookies On
    Require valid-user
```

Ajoutez en dessous de Require valid-user :

```
SetEnvIf Cookie "user_name=(^[^;]+)" REMOTE_USER=$1
```

---

Afin d'obtenir:

```
Alias /cacti /srv/eyesofnetwork/cacti
<Directory /srv/eyesofnetwork/cacti>
  AuthType Basic
  AuthName "Restricted"
  AuthFormAuthoritative On
  AuthFormMySQLSocket /var/lib/mysql/mysql.sock
  AuthFormMySQLUsername eonweb
  AuthFormMySQLPassword root66
  AuthFormMySQLDB eonweb
  AuthFormMySQLTableSID sessions,users,groupright
  AuthFormMySQLFieldUID sessions.user_id
  AuthFormMySQLTableSIDCondition "`sessions`.`session_id`=$session_id AND
`sessions`.`user_id`=$user_id AND `users`.`group_id`=$group_id AND `groupright`.`group_id`=$group_id
AND `groupright`.`tab_6`='1'"
  AuthFormPageLogin /login.php
  AuthFormSessionCookies On
  Require valid-user
  SetEnvIf Cookie "user_name=(^[^;]+)" REMOTE_USER=$1

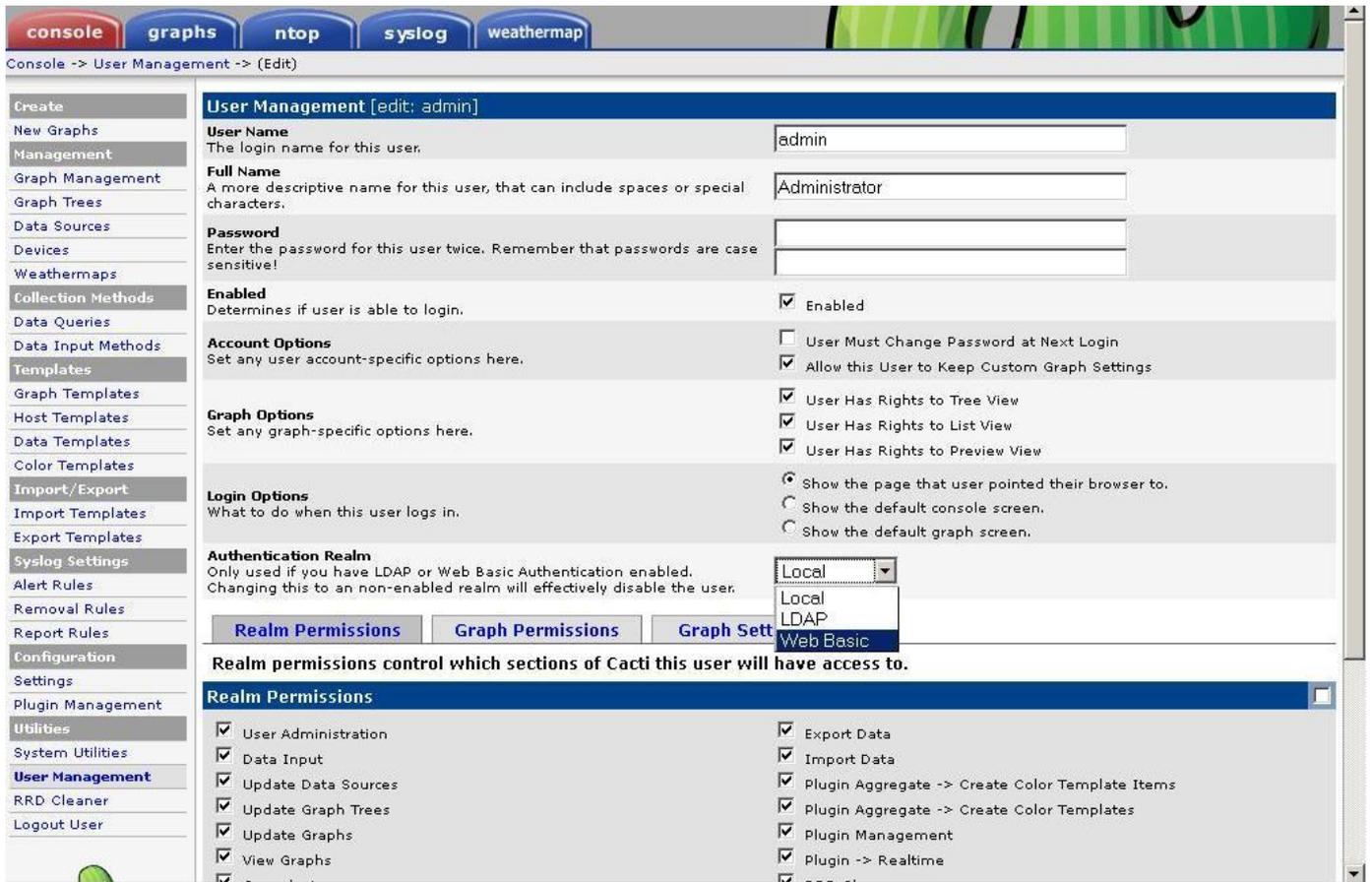
  <Files ~ (weathermap-cacti-plugin.php|weathermap-eonweb-
plugin.php|graph_image.php|ntop.php|syslog.php)>
```

...  
Une fois ceci fait redémarrez le service httpd:

```
systemctl restart httpd
```

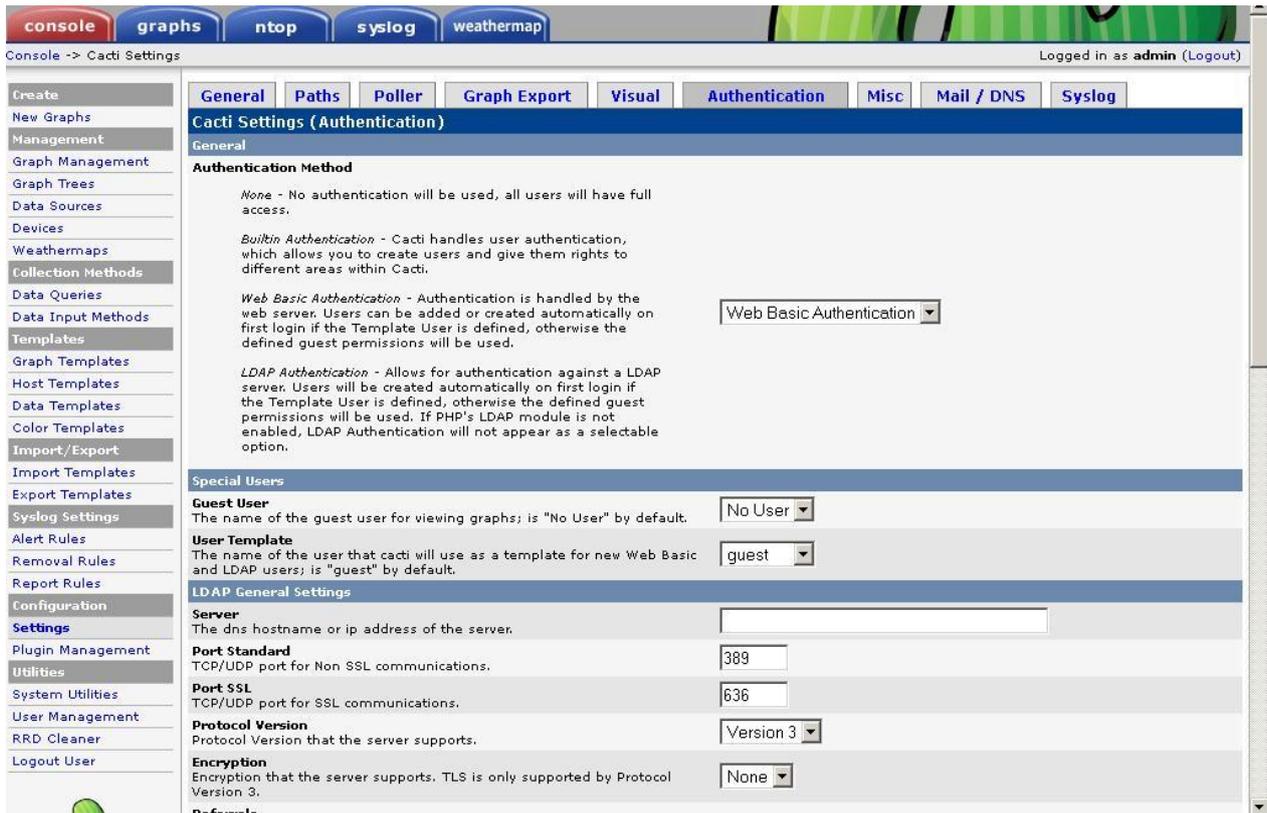
Loguez-vous au portail eonweb en admin et allez dans Cacti.

Une fois dans l'application cliquez sur le bandeau de gauche sur « User Management » afin d'avoir ceci et sélectionnez le compte « admin » :



Passez la variable « Authentication Realm » de « Local » à « Web Basic ».  
Faites « Save » en bas de page.

Allez ensuite dans la section « Settings » du bandeau de gauche, puis sélectionnez à l'issue l'onglet « Authentication » :



Passez la variable « Authentication Method » de « None » à « Web Basic », vérifiez aussi que « User Template » soit sur « guest ».

Voilà ! Maintenant seule la personne logué en « admin » à eonweb sera autorisé à se connecter à Cacti, voir les graphes, etc...

Si à cette page vous activez le « User Template » à « guest » (comme fait ci-dessus), les autres personnes logués à eonweb (hors admin) voulant accéder à Cacti auront un compte Cacti automatiquement créée basé sur les droits de « guest ». N'oubliez donc pas de vérifier les droits attribués au compte guest.

Voici un exemple de droits donné au « template » guest. Notez au passage que le compte test présent a été créé par Cacti lors d'essais de ma part. J'attire votre attention sur le fait que « Enabled » doit être coché !

User Management						Add	
Search: <input type="text"/>						Go	Clear
<< Previous		Showing Rows 1 to 3 of 3 [1]			Next >>		
User Name**	Full Name	Enabled	Realm	Default Graph Policy	Last Login	<input type="checkbox"/>	
admin	Administrator	Yes	Web Basic	ALLOW	Tuesday, July 23, 2013 10:26:43	<input type="checkbox"/>	
guest	Guest Account	Yes	Local	ALLOW	N/A	<input type="checkbox"/>	
test	Guest Account	Yes	Web Basic	ALLOW	Tuesday, July 23, 2013 10:26:18	<input type="checkbox"/>	
<< Previous		Showing Rows 1 to 3 of 3 [1]			Next >>		
Choose an action:						Delete	Go

The screenshot shows the Cacti web interface for user management. The user 'guest' is being configured. The 'Enabled' checkbox is checked and circled in red. The 'Authentication Realm' is set to 'Local'. The 'Realm Permissions' section is expanded, showing a list of permissions with checkboxes.

Realm Permissions	Graph Permissions	Graph Settings
<input type="checkbox"/> User Administration		
<input type="checkbox"/> Data Input		
<input type="checkbox"/> Update Data Sources		
<input type="checkbox"/> Update Graph Trees		
<input type="checkbox"/> Update Graphs		
<input checked="" type="checkbox"/> View Graphs		
<input type="checkbox"/> Export Data		
<input type="checkbox"/> Import Data		
<input type="checkbox"/> Plugin Aggregate -> Create Color Template Items		
<input type="checkbox"/> Plugin Aggregate -> Create Color Templates		
<input type="checkbox"/> Plugin Management		
<input type="checkbox"/> Plugin -> Realtime		

Tous les utilisateurs que vous allez créer par l'interface onweb « Administration -> Généralités -> Utilisateurs » pourront donc tous consulter Cacti mais avec les droits de guest (sauf admin).

Mais comme chaque user sera créé aussi dans Cacti vous pourrez affiner les droits d'accès aux cartes Weathermap ou autre !

A vous d'en faire un bon usage.

---

## **- Prenons le cas de Nagios :**

Effectuez le même genre de manipulation pour le fichier /etc/httpd/conf.d/nagios.conf que pour le fichier de Cacti.

Donc au lieu de :

```
ScriptAlias /nagios/cgi-bin "/srv/eyesofnetwork/nagios/sbin"
<Directory "/srv/eyesofnetwork/nagios/sbin">
    Options ExecCGI
    AuthType Basic
    AuthName "Restricted"
    AuthFormAuthoritative On
    AuthFormMySQLSocket /var/lib/mysql/mysql.sock
    AuthFormMySQLUsername eonweb
    AuthFormMySQLPassword root66
    AuthFormMySQLDB eonweb
    AuthFormMySQLTableSID sessions,users,groupright
    AuthFormMySQLFieldUID sessions.user_id
    AuthFormMySQLTableSIDCondition "`sessions`.`session_id`=$session_id AND
`sessions`.`user_id`=$user_id AND `users`.`group_id`=$group_id AND `groupright`.`group_id`=$group_id
AND (`groupright`.`tab_2`='1' OR `groupright`.`tab_5`='1' OR `groupright`.`tab_6`='1')"
```

```
    AuthFormPageLogin /login.php
    AuthFormSessionCookies On
    Require valid-user

# monitoring file access
<Files ~ (cmd.cgi|tac.cgi|status.cgi|statusmap.cgi|outages.cgi|extinfo.cgi)>
```

Ajouter la ligne « SetEnvIf Cookie "user\_name=(^[^;]+)" REMOTE\_USER=\$1 » comme suit :

```
ScriptAlias /nagios/cgi-bin "/srv/eyesofnetwork/nagios/sbin"
<Directory "/srv/eyesofnetwork/nagios/sbin">
    Options ExecCGI
    AuthType Basic
    AuthName "Restricted"
    AuthFormAuthoritative On
    AuthFormMySQLSocket /var/lib/mysql/mysql.sock
    AuthFormMySQLUsername eonweb
    AuthFormMySQLPassword root66
    AuthFormMySQLDB eonweb
    AuthFormMySQLTableSID sessions,users,groupright
    AuthFormMySQLFieldUID sessions.user_id
    AuthFormMySQLTableSIDCondition "`sessions`.`session_id`=$session_id AND
`sessions`.`user_id`=$user_id AND `users`.`group_id`=$group_id AND `groupright`.`group_id`=$group_id
AND (`groupright`.`tab_2`='1' OR `groupright`.`tab_5`='1' OR `groupright`.`tab_6`='1')"
```

```
    AuthFormPageLogin /login.php
    AuthFormSessionCookies On
    Require valid-user
    SetEnvIf Cookie "user_name=(^[^;]+)" REMOTE_USER=$1
# monitoring file access
<Files ~ (cmd.cgi|tac.cgi|status.cgi|statusmap.cgi|outages.cgi|extinfo.cgi)>
```

Sauvegardez/quittez.

Effectuez la même manipulation concernant le fichier `/etc/httpd/conf.d/thruk.conf` !!!

Une fois ceci fait redémarrez le service httpd:

```
systemctl restart httpd
```

Voilà !

Un utilisateur « toto » logué en « toto » sur eonweb aura accès à Nagios et ne verra (comme dans notre cas) : RIEN !

C'est à ce moment que la notion de « contact » Nagios prend son sens.

Un user eonweb non déclaré en tant que contact (ou aussi en faisant partie d'un contact group)  
A des host/services, ne verra aucun de ceux-ci !

Vous noterez du coup que sur la page Nagios affichée vous affichera bien « logué en « toto » » ou « test » dans ce cas d'exemple :



Vous voyez bien que « test » n'accède à rien sous Nagios !

---

- **Prenons le cas eonweb :**

Si vous voulez activer les cookies de manière globale pour tous les « sites », de la même manière que précédemment éditez le fichier /etc/httpd/conf.d/eonweb.conf et modifiez les lignes :

```
SetEnvIf Cookie "user_name=([^;]+)" REMOTE_USER=$admin
#SetEnvIf Cookie "user_name=([^;]+)" REMOTE_USER=$1
```

Pour obtenir la même chose que ceci:

```
# --- REMOTE_USER
<Directory /srv/eyesofnetwork>
    #SetEnvIf Cookie "user_name=([^;]+)" REMOTE_USER=$admin
    SetEnvIf Cookie "user_name=([^;]+)" REMOTE_USER=$1
</Directory>
# --- END REMOTE_USER

<Directory /srv/eyesofnetwork/eonweb>
    Options -Indexes
</Directory>

<Directory ~ /srv/eyesofnetwork/eonweb/(include|cache|bin|module)>
    AuthType Basic
    AuthName "Restricted"
    AuthFormAuthoritative On
```

Et redémarrez aussi le service httpd.

Pour chaque « Site », modifiez donc ces quelques lignes pour activer les cookies !

---

## 9. Configuration du rsyslog.conf pour recevoir les logs d'autres hosts :

Par défaut le plugin Syslog de Cacti, qui s'appuie sur le service Rsyslog, n'affiche que les logs locaux. Pour la prise en charge des logs d'autres hosts par IP faites ce qui suit :

- Editez en console le fichier /etc/rsyslog.conf (vi /etc/rsyslog.conf)
- Ajoutez sous la ligne « \$ModLoad imuxsock » les lignes :

```
# Provides UDP syslog reception
$ModLoad imudp
$UDPServerRun 514
# Provides TCP syslog reception
$ModLoad imtcp
$InputTCPServerRun 514
```

Sauvegardez et quittez (Faites « echap » puis :wq ! ou :x !).

Une fois fait redémarrez les services Rsyslog.

```
systemctl restart rsyslog
```

Vous pourrez désormais recevoir les logs de vos autres équipements !

---

## 10. Mise en place de gnokii pour activer les notifications par SMS

Ce qui suit est tiré du forum EyesOfNetwork, plus particulièrement d'un post de l'utilisateur « [jamalmellal](#) », Le mérite lui en revient !

Installation de gnokii, site officiel : <http://www.gnokii.org/downloads.shtml>

La version 6.14 a été testée sur EoN v2.2.

Via un utilitaire type winscp, copiez le fichier téléchargé sur EoN

Mettez-vous en console root sur EoN avant de poursuivre...

Après copie sur EoN du fichier, extraire les archives avec les commandes suivantes

```
tar -xzf gnokii-0.6.14.tar.gz
```

```
tar -xjf gnokii-0.6.14.tar.bz2
```

Déplacez-vous sur le dossier gnokii-0.6.14 :

```
cd gnokii-0.6.14
```

Configuration :

```
./configure
```

Installation :

```
gmake  
gmake install  
gmake install-docs
```

Copiez le fichier gnokiirc situé dans le dossier Docs/sample/ dans le dossier racine /etc

il faut ouvrir le fichier gnokiirc (utilisez la commande « vi » par exemple) copié et configurer votre gsm

```
port = /dev/ttyACM0  
model = AT  
connection = serial
```

---

NB : le port peut changer en fonction de votre gsm.

Vous pouvez maintenant faire un test d'envoi de sms avec root comme suit :

```
echo 'Test avec sms gnokii' | gnokii --config /etc/gnokiirc --sendsms +212555555555
```

Si vous passez en user nagios (su nagios) vous aurez un problème de droit d'accès au dossier lock donc donnez le droit au user nagios, soit en l'ajoutant au groupe uucp, soit en changeant le propriétaire du dossier...(était root devient nagios)

Maintenant mettez-vous en user nagios (su nagios) et envoyer un sms de test

```
echo 'Test avec sms gnokii' | gnokii --config /etc/gnokiirc --sendsms +212555555555
```

Si c'est réussi, configurez lilac en ajoutant une commande nommée notify-host-by-sms, par exemple (pareil pour les services sur le principe).

La commande sera comme suit (command\_line):

```
/usr/bin/printf "Alert Nagios $NOTIFICATIONTYPE$ : Host $HOSTALIAS$ is $HOSTSTATE$" | /usr/local/bin/gnokii --sendsms $CONTACTPAGER$
```

N'oubliez pas de renseigner sur « Nagios configuration » (ex-lilac) le numéro de gsm du contact devant recevoir les sms (pager)

NB : vous pouvez enrichir la commande pour recevoir d'autres informations par sms

---

## 11. Modification du backend Livestatus

### 11.1. Introduction

EO N n'utilise pas `ndo` mais `mkLivestatus`. La différence principale est que `mkLivestatus` s'implémente au cœur de Nagios. Il n'accède donc à aucune base de donnée ni à aucun fichier.

C'est l'utilitaire `unixcat` (fourni par `mkLivestatus`) qui permet (via le langage LQL Livestatus Query Language ) d'interroger Nagios.

L'interrogation d'un serveur "satellite" (on parlera de backend) depuis un serveur central, n'est pas possible nativement avec `mkLivestatus`. Mais cela ne veut pas dire que c'est impossible. Il suffit d'installer et de configurer `xinetd`. Dans cet exemple il sera supposé que vous avez un repository local ou que vous êtes connectés au web (pour effectuer le `yum` correctement).

### 11.2. Configuration d'un backend

Sur le serveur « satellite », on commence par installer `xinetd` :

```
yum install xinetd
```

puis on crée le service à lancer dans `/etc/xinetd.d/central` (par exemple) :

```
service livestatus_sat1
{
  disable = no
  type = UNLISTED
  port = 6700
  socket_type = stream
  protocol = tcp
  wait = no
  user = nagios
  flags = NOLIBWRAP
  server = /srv/eyesofnetwork/mk-livestatus-1.2.0p1/bin/unixcat
  server_args = /srv/eyesofnetwork/nagios-3.4.1/var/log/rw/live
}
```

Ici on ouvre donc un socket TCP (6700) derrière lequel sera lancé l'utilitaire `unixcat`.

Il est nécessaire de redémarrer `xinetd` :

```
systemctl restart xinet.d
```

### 11.3. Déclaration des backends sur le serveur central

Il existe 2 possibilités pour déclarer les backends sur le serveur central :

- manuellement dans le fichier `/srv/eyesofnetwork/nagvis/etc/nagvis.ini.php` ;
- dans l'interface web de Nagvis

---

## 11.4. Déclaration manuelle

Ouvrez le fichier `/srv/eyesofnetwork/nagvis/etc/nagvis.ini.php` avec « vi ».

Déclarez le backend:

```
[backend_sat1]
backendtype="mklivestatus"
socket="tcp:sat1:6700"
```

Attention, le nom de machine "sat1" doit être résolu. Si ce n'est pas le cas, remplacer "sat1" par son adresse IP.

## 11.5. Déclaration via Nagvis

Se rendre dans l'interface web de Nagvis dans « option / manage backend », section « Add backend »

The screenshot shows the 'Add backend' form in Nagvis. The fields are as follows:

backendid	sat1
backendtype	mklivestatus
socket	tcp:sat1:6700
timeout	500
statushost	
htmlcgi	
custom_1	
custom_2	
custom_3	

Save

Cliquez sur « Save ».

## 11.6. Ajouter un host ou service avec ce backend

Il est maintenant possible d'afficher l'état des machines du backend sur une carte du serveur central.

Pour cela ouvrir une carte sous Nagvis. Faites ensuite « Edit Map -> Add Icon -> Host », placez l'icône à l'endroit voulu sur la carte et choisir le bon backend dans "backend\_id".

---

## Create Object

X

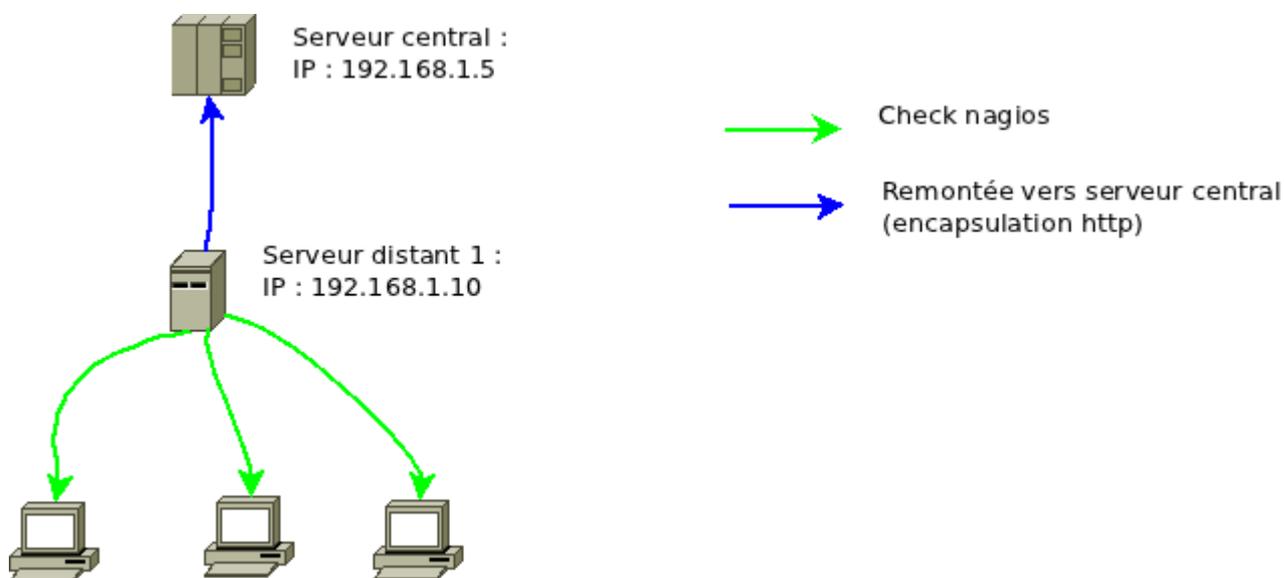
host_name	<input type="text"/>
x	818
y	176
z	<input type="checkbox"/> 10
backend_id	<input checked="" type="checkbox"/> live_1
view_type	<input type="checkbox"/>
iconset	<input type="checkbox"/>
context_menu	<input type="checkbox"/>
context_template	<input type="checkbox"/>
exclude_members	<input type="checkbox"/>
exclude_member_states	<input type="checkbox"/>

- live\_1
- ndomy\_1
- merlinmy\_1
- demo
- >>> Specify other

Ici un exemple d'affichage avec un backend. Par la suite, les hosts du serveur "satellite" s'affichent correctement dans le champ host\_name.

## 12. Configuration de GED pour une architecture distribuée :

GED (Generic Event Dispatcher) permet de faire remonter les événements d'un site distant, vers un site central. Voici un exemple « d'architecture » :



Site Web de GED : <http://generic-ed.sourceforge.net/>

Pour pouvoir afficher les remontés des changements d'états des hôtes/services (dans la vue événements actifs) du serveur distant dans le central une reconfiguration des fichiers ged va être nécessaire.

### 12.1. Configuration du serveur central

**Editez le fichier /srv/eyesofnetwork/ged/ged.cfg:**

Il convient de renseigner la plage d'IPs autorisée à se synchroniser avec le serveur central.

Il faut modifier pour cela la variable "**allow\_sync**". Par défaut tout est autorisé.

Donc dans notre exemple : `allow_sync = 192.168.1.10`

### 12.2. Configuration du serveur distant

**Editez le fichier /srv/eyesofnetwork/ged/ged.cfg:**

Il faut autoriser le serveur central à contacter le serveur distant.

---

La variable "**allow\_request\_from**"

Ce paramètre **allow\_request\_from** n'est pas obligatoire pour effectuer les synchro. Ce paramètre permet simplement de spécifier quelles plages d'adresses distantes peuvent effectuer des opérations de push et drop depuis gedq ou tout simplement avoir une lisibilité complète sur l'ensemble des événements en queue y compris des événements dont la plage en question n'est pas la source (par défaut si A et B poussent vers C, A n'a pas de visu sur les événements de B en C et B n'a pas de visu sur les événements de A en C).

Dans cet exemple, bien qu'optionnel : `allow_request_from 192.168.1.5`

Ensuite, il faut dé-commenter la dernière ligne : « **include /srv/eyesofnetwork/ged/etc/gedt.cfg** »

Ainsi, le serveur distant saura vers quel serveur il doit relayer les events(voir ci-dessous).

### **Editez le fichier /srv/eyesofnetwork/ged/gedt.cfg**

Il s'agit de définir vers quel serveur et sur quel port on doit "relayer" les events (notre serveur central).

Dans le « bloque <relay\_to> » il faut modifier la variable "connect"

Donc dans notre exemple : `connect 1192.168.1.5:2403`

### **Editez le fichier /srv/eyesofnetwork/ged/gedmysql.cfg**

Mettre le "`ttl_sync_queue`" 300 (= mise a jour de la table ged 5 min après).



Le paramètre **ttl\_sync\_queue** 300 signifie qu'un événement en attente de synchro ne persistera en queue de synchro que 5 min. Passé ce délai, le relai ne sera pas effectué et la synchro de l'événement annulée.

Choisissez donc une valeur adéquate.

## **12.3. Redémarrage du processus ged**

Pour que les modifications soient prises en compte il vous faut redémarrer le service ged sur les deux serveurs.

Placez-vous au niveau de la section « Administration -> Généralités -> Processus » du portail eonweb.

The screenshot shows the 'Eyes Of Network' interface. The main title is 'Gestion des processus'. On the left, there is a sidebar with navigation options: 'Tableaux de bord', 'Disponibilités', 'Capacité', 'Production', 'Rapports', 'Administration', and 'Aide'. The main content area displays a table of processes. The table has columns for 'processus', 'status', 'PID', and 'actions'. The 'Ged agent' process is highlighted in green, and its 'Redémarrer' button is circled in red. Below the table, there is a footer that reads 'EyesOfNetwork produit sous licence GPL2, sponsorisé par AXIANS'.

processus	status	PID	actions
Nagios	UP	25572 25569 25568 25565 25563 25562 25559 25556 25555 25553 25552 25549 25495 25485 25473 25471 25469 25467 25461 25452 25450 25449 25409 25407 25386 25377 25376 25374 25358 25356 25355 25352 25349 25348 25346 25345 1200	Stopper, Redémarrer, Recharger, Vérifier
Ged agent	UP	22366	Stopper, Redémarrer, Recharger
SNMP agent	UP	17408	Stopper, Redémarrer, Recharger
SNMP trap agent	UP	1198	Stopper, Redémarrer, Recharger
SNMP trap traductor	UP	612	Stopper, Redémarrer, Recharger

En face de "Ged agent" cliquer sur **restart**

## 12.4. Vérifications :

### Au niveau de la table Mysql :

Sur le serveur secondaire, on peut vérifier que les informations se trouvent bien dans la table nagios\_queue\_XXX (ou XXX sera remplacé par active, sync, history) de la bdd GED.

Nous utiliserons ici nagios\_queue\_history pour accéder à des infos, si vous cherchez vraiment les infos de synchronisation préférez sync.

Se connecter root (mdp : root66) sur la table mysql :

```
mysql ged -u root -p
```

Une fois connecté :

```
select * from nagios_queue_history ;
```

Si les informations sont bien présentes, faites la même chose sur le serveur central, vous devriez voir les informations du serveur secondaire. A titre d'information, les valeurs o\_, l\_, r\_, m\_, correspondent respectivement à « original timestamp », « last occurrence timestamp », « last reception timestamp », « last user meta timestamp ».

**ATTENTION :** les informations ne sont écrites dans cette table que lorsque un check change d'état (exemple : ok → critique ou warning → ok). Pour tester vous pouvez par exemple modifier un check pour changer son état ou faire tomber un équipement...

## Au niveau des logs :

Il est possible de modifier le niveau de log dans le fichier "ged.cfg".

Par exemple, modifier la variable : « syslog\_level 16 » vous permettra de voir les push, drop ...

Tout est stocké dans /var/log/message.

Exemple :

```
Oct 23 04:04:51 s_sys@eon-secondaire ged: MYSQL backend push sync queue on packet type "1"
[192.168.1.5] : created occurrence
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend peek active queue on packet type "1" [] : 1
result(s)
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend push sync queue on packet type "1" [192.168.1
.5] : created occurrence
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend drop active queue on packet type "1" [] : dropped
1 record(s)
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend peek active queue on packet type "1" [] : 1
result(s)
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend push sync queue on packet type "1" [192.168.1
5] : created occurrence
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend drop active queue on packet type "1" [] : dropped
1 record(s)
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend peek active queue on packet type "1" [] : 0
result(s)
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend push sync queue on packet type "1" [192.168.1
.5] : created occurrence
Oct 23 04:04:52 s_sys@eon-secondaire ged: MYSQL backend drop active queue on packet type "1" [] : dropped
0 record(s)
```

## 12.5. Visualisation dans l'interface « Eonweb », onglet disponibilités :

### Dans les évènements actifs :

Sur le serveur central, rendez-vous dans la section « Disponibilité -> Evènements -> Evènements actifs ».

ALL	EQUIPMENT	SERVICE	STATE	OWNER	DESCRIPTION	ORIGINAL-TIME	LAST-TIME	OCCURENCES
<input type="checkbox"/>	<a href="#">Windows_2003_secondaire</a>	<a href="#">memory</a>			ERROR: Description/Type table : No response from remote host 172.17.212.3.	Oct 24, 2011 10:15 AM	Oct 24, 2011 10:15 AM	1
<input type="checkbox"/>	<a href="#">Windows_2003_secondaire</a>	<a href="#">HOST DOWN</a>			(Host Check Timed Out)	Oct 24, 2011 10:14 AM	Oct 24, 2011 10:14 AM	1
<input type="checkbox"/>	<a href="#">Windows_2003_secondaire</a>	<a href="#">uptime</a>			CRITICAL: Systemuptime 0:02:44.54.	Oct 24, 2011 10:10 AM	Oct 24, 2011 10:12 AM	3
<input type="checkbox"/>	<a href="#">Windows_2003_secondaire</a>	<a href="#">uptime</a>			(No output returned from plugin)	Oct 24, 2011 10:09 AM	Oct 24, 2011 10:09 AM	1
<input type="checkbox"/>	<a href="#">Windows_2003_secondaire</a>	<a href="#">processor</a>			ERROR: Description table : No response from remote host 172.17.212.3.	Oct 24, 2011 10:08 AM	Oct 24, 2011 10:14 AM	4
ALL	EQUIPMENT	SERVICE	STATE	OWNER	DESCRIPTION	ORIGINAL-TIME	LAST-TIME	OCCURENCES

Ici, on peut voir par exemple, un redémarrage d'un serveur windows 2003 (nommé "Windows 2003 Secondaire" sur le serveur EoN secondaire). Il apparaît donc down sur le serveur central.

### Dans les évènements résolus :

Sur le serveur central, rendez-vous dans la section « Disponibilité -> Evènements -> Evènements résolus ».

ALL	EQUIPMENT	SERVICE	STATE	OWNER	DESCRIPTION	ORIGINAL-TIME	LAST-TIME	OCCURENCES
<input type="checkbox"/>	Windows_2003_secondaire	memory			Physical Memory: 41%used(422MB/1022MB) Virtual Memory: 15%used(361MB/2469MB) (80%) : OK	Oct 24, 2011 10:22 AM	Oct 24, 2011 10:22 AM	1
<input type="checkbox"/>	Windows_2003_secondaire	processor			2 CPU, average load 9.5% 80% : OK	Oct 24, 2011 10:22 AM	Oct 24, 2011 10:22 AM	1
<input type="checkbox"/>	Windows_2003_secondaire	system			System Time OK - 10-24-2011, 10:21:34	Oct 24, 2011 10:21 AM	Oct 24, 2011 10:21 AM	1
<input type="checkbox"/>	Windows_2003_secondaire	partitions			All selected storages (90%) : OK	Oct 24, 2011 10:20 AM	Oct 24, 2011 10:20 AM	1
<input type="checkbox"/>	Windows_2003_secondaire	HOST UP			PING OK - Paquets perdus = 0%, RTA = 0.98 ms	Oct 24, 2011 10:19 AM	Oct 24, 2011 10:19 AM	1

Une fois le serveur Windows redémarré, il apparaît bien dans les évènements résolus. A noter : cet évènement dit "résolu" n'apparaît donc plus dans la liste des évènements actifs.

Il est possible d'utiliser les filtres pour n'afficher par exemple que les évènements d'un équipement.

---

## 13. Conclusion :

Ces exemples ne sont qu'une partie de ce que l'on peut faire afin de vous aider à franchir le pas.  
D'autres points devraient peut être faire partie de cette aide, merci à la communauté de bien vouloir participer en faisant des remontés de tests ou autres !

Document rédigé par :  
FERNANDEZ Sébastien,  
au profit de la communauté EyesOfNetwork