

Nagios

Configuration d'un Client Linux (Debian/Ubuntu) pour Nagios Log Server

Prérequis

- Disposer d'une VM Debian ou Ubuntu (le "Client").
 - Avoir les privilèges **Root** ou un accès `sudo` .
 - Avoir une connectivité réseau avec le serveur Nagios (Ping fonctionnel).
-

Étape 1 : Installation de Rsyslog

Rsyslog est le service qui va collecter les logs locaux et les expédier. Assurez-vous qu'il est installé et à jour.

Mise à jour du système :

Bash

```
sudo apt update
```

Installation de Rsyslog :

Bash

```
sudo apt install rsyslog -y
```

Activer le service pour qu'il démarre automatiquement au boot :

Bash

```
sudo systemctl enable rsyslog  
sudo systemctl start rsyslog
```

Étape 2 : Configuration de l'envoi des logs

Choisissez **UNE** des deux méthodes ci-dessous (la méthode automatique est recommandée pour Nagios Log Server).

Méthode A : Configuration Automatique (Via Script Nagios)

Cette méthode télécharge un script depuis votre serveur Nagios qui configure tout automatiquement.

1. Télécharger le script de configuration (remplacez `IP_DU_SERVEUR_NAGIOS` par la vraie IP de votre serveur Nagios) :

Bash

```
curl -sS -O http://IP_DU_SERVEUR_NAGIOS/nagioslogserver/scripts/setup-linux.sh
```

2. Exécuter le script avec les privilèges root (le port standard pour Nagios Log Server est souvent **5544**) :

Bash

```
# Remplacez IP_DU_SERVEUR_NAGIOS par l'IP de votre serveur Nagios
sudo bash setup-linux.sh -s IP_DU_SERVEUR_NAGIOS -p 5544
```

Méthode B : Configuration Manuelle (Via fichier conf)

Utilisez cette méthode si vous voulez comprendre ce qui se passe ou si le script ne marche pas.

1. Ouvrir le fichier de configuration :

Bash

```
sudo nano /etc/rsyslog.d/99-nagioslog.conf
```

(Note : Il est plus propre de créer un nouveau fichier dans `/etc/rsyslog.d/` plutôt que de modifier le fichier principal `/etc/rsyslog.conf`, mais les deux fonctionnent).

- Ajouter la ligne suivante : *Remplacez 192.168.1.100 par l'IP de votre serveur Nagios.*
Note sur les arobases : @@ signifie TCP (plus fiable), @ signifie UDP (plus rapide).

Bash

```
# Envoi de tous les logs (*.*) vers le serveur en TCP sur le port 5544
*.* @@192.168.1.100:5544
```

Si votre serveur Nagios écoute sur le port standard syslog, mettez 514. Si c'est une input spécifique Nagios Log Server, c'est souvent 5544.

- Sauvegarder (Ctrl+O, Entrée) et quitter (Ctrl+X).
- Redémarrer le service pour appliquer les changements :

Bash

```
sudo systemctl restart rsyslog
```

Étape 3 : Vérification

- Vérifier que Rsyslog tourne sans erreur :

Bash

```
sudo systemctl status rsyslog
```

- Générer un log de test manuellement sur le client :

Bash

```
logger "Test de connexion vers Nagios Log Server depuis mon Client Debian"
```

- Sur l'interface Web de Nagios Log Server :**

- Allez dans le menu **Dashboards**.
- Regardez les logs récents. Vous devriez voir votre message de test apparaître.
- Vous pouvez filtrer par l'adresse IP de votre client (utilisez `ip a` sur le client pour confirmer son IP).

Documentation Technique : Configuration Client Rsyslog (Haute Disponibilité)

1. Contexte et Architecture

Cette procédure décrit l'installation et la configuration de l'agent **Rsyslog** sur un client Linux (Debian/Ubuntu). Dans une optique de **Haute Disponibilité (HA)**, le client ne communique pas directement avec les nœuds de stockage. Il envoie ses logs vers un **Répartiteur de Charge (Nginx)** qui se charge de distribuer le flux vers le cluster Nagios Log Server.

Flux de données : [Client Linux] → [Load Balancer Nginx] → [Cluster Nagios (Nœud A / Nœud B)]

2. Prérequis

- Une machine virtuelle Ubuntu ou Debian (Le Client).
- Accès privilèges **Root** ou `sudo`.
- Connectivité réseau établie vers le Load Balancer.
- **Adresse IP du Load Balancer (VIP) :** A RENSEIGNER (ex: 192.168.1.200)

3. Installation du service Rsyslog

S'assurer que le système est à jour et que le service est présent.

Bash

```
# Mise à jour des dépôts
sudo apt update

# Installation de rsyslog (s'il n'est pas déjà présent)
sudo apt install rsyslog -y

# Activation du service au démarrage et lancement immédiat
sudo systemctl enable rsyslog
sudo systemctl start rsyslog
```

4. Configuration de l'envoi des logs (Mode HA)

Nous allons créer un fichier de configuration dédié pour ne pas altérer la configuration par défaut du système.

1. Créer le fichier de configuration :

Bash

```
sudo nano /etc/rsyslog.d/99-nagios-ha.conf
```

2. Ajouter la directive suivante : ⚠ *Attention : Remplacez `IP_LOAD_BALANCER` par l'adresse IP de votre serveur Nginx, pas celle du serveur Nagios final.*

Bash

```
# Transfert de tous les logs (*.*) vers le Load Balancer via TCP (@@)
# Syntaxe : *.* @[IP_DU_LOAD_BALANCER]:[PORT]
*.* @@192.168.1.200:5544
```

Note technique :

- `*.*` : Tous les services, tous les niveaux de sévérité.
- `@@` : Utilisation du protocole **TCP**. C'est crucial pour la fiabilité (garantit que les logs arrivent bien au Load Balancer).
- `5544` : Port d'écoute standard pour les inputs Nagios Log Server.

3. Sauvegarder et quitter : (CTRL+O, Entrée, CTRL+X)

4. Redémarrer le service pour appliquer les changements :

Bash

```
sudo systemctl restart rsyslog
```

5. Vérification et Tests

1. Vérifier l'état du service :

Bash

```
sudo systemctl status rsyslog
```

Le statut doit être "Active (running)".

2. Générer un log de test : Envoyez un message manuel pour vérifier que la chaîne complète (Client > Nginx > Nagios) fonctionne.

Bash

```
logger "Test de Haute Disponibilité : Connexion via Nginx Load Balancer OK"
```

3. Validation finale : Connectez-vous sur l'interface Web de **Nagios Log Server** (Dashboard). Vous devriez voir apparaître le message ci-dessus. Si le message apparaît, cela confirme que :

1. Le client a bien envoyé le log au Nginx.
2. Le Nginx a bien redirigé le log vers un nœud Nagios disponible.