

Bilel RAHMOUNI

# Étude et mise en œuvre d'une solution de supervision réseau

---

## Objectifs :

- Découvrir et comparer des solutions de supervision existantes (Nagios, Cac2, etc.).
- Comprendre les concepts de base d'un système de supervision (gestionnaire, agent, métriques).
- Installer et configurer un logiciel de supervision pour collecter des données en temps réel.
- Tester le fonctionnement avec des requêtes simples et interpréter les résultats.

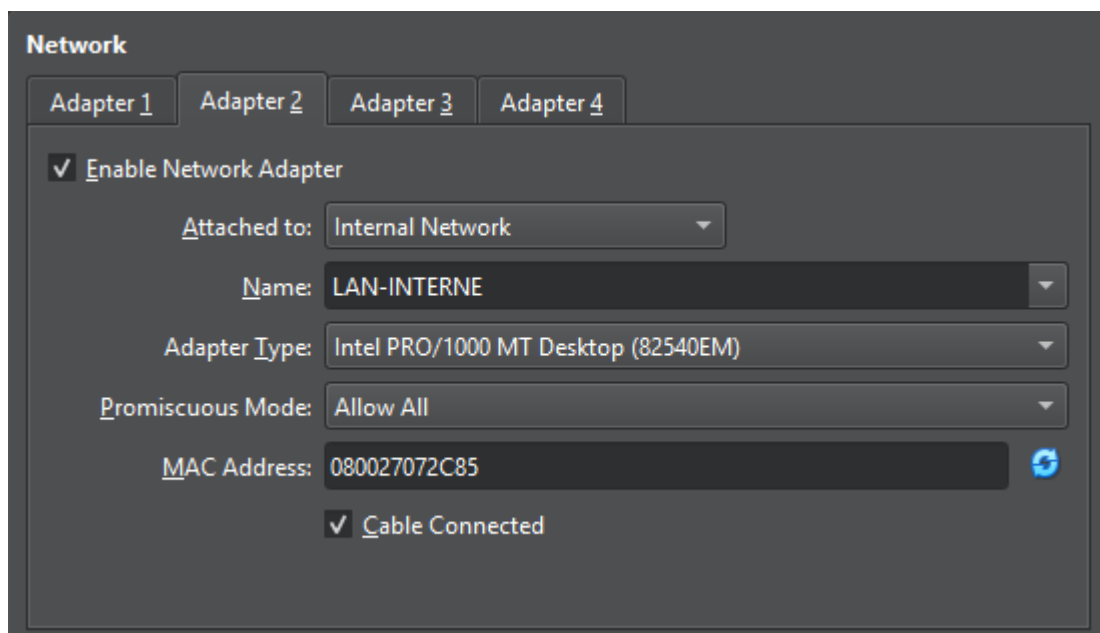
# 1. Présentation de l'infrastructure mise en place

## Architecture

- **Machine 1 : Gestionnaire (Cyclop OS)**
  - Rôle : Hôte principal pour la supervision.
  - Logiciel installé : Zabbix Server.
  - Adresse IP : 192.168.56.101.
- **Machine 2 : Agent (Kali Linux)**
  - Rôle : Hôte supervisé.
  - Service installé : Apache.
  - Adresse IP : 192.168.56.102.

## Connexion réseau

- Type : Connexion interne entre les deux machines virtuelles via un réseau interne (VirtualBox ou VMware).
- Configuration :
  1. Accédez aux paramètres réseau de chaque machine virtuelle.
  2. Réglez l'adaptateur réseau sur le mode **Réseau interne**.
  3. Donnez un nom identique au réseau interne (ex. : LAN-INTERNE).



- Test de connectivité :
  1. Démarrez les deux machines virtuelles.

Configurez les adresses IP manuellement :

```
# Sur Cyclop OS
sudo ifconfig eth0 192.168.56.101 netmask 255.255.255.0

# Sur Kali Linux
sudo ifconfig eth0 192.168.56.102 netmask 255.255.255.0
```

Cyclop OS:

```
rahmonex@rahmonex-pc:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:05:78:f5 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 86104sec preferred_lft 86104sec
    inet6 fd00::e5f4:f64c:a15e:bd93/64 scope global temporary dynamic
        valid_lft 86393sec preferred_lft 14393sec
    inet6 fd00::d5f5:a959:be4e:f957/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86393sec preferred_lft 14393sec
    inet6 fe80::acd4:b32b:978:626d/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:07:2c:85 brd ff:ff:ff:ff:ff:ff
rahmonex@rahmonex-pc:~$ sudo ifconfig enp0s8 192.168.56.101 netmask 255.255.255.0
```

Kali Linux:

```
(rahmonex@vbox)~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:13:4a:6e brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:be:43:bf brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:febe:43bf/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
(rahmonex@vbox)~$ sudo ifconfig eth1 192.168.56.102 netmask 255.255.255.0
```

2. Testez la connectivité réseau avec la commande **ping** :

```
ping 192.168.56.102 # Depuis Cyclop OS
ping 192.168.56.101 # Depuis Kali Linux
```

Ping de Kali Linux vers Cyclop OS :

```
(rahmonex@vbox)~$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data:
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=3.26 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.876 ms
^C
— 192.168.56.101 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1004ms
rtt min/avg/max/mdev = 0.876/2.067/3.258/1.191 ms
```

*Ping de Cyclop OS vers Kali Linux :*

```
rahmonex@rahmonex-pc:~$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=1.66 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=16.9 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.803 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.564 ms
^C
--- 192.168.56.102 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3294ms
rtt min/avg/max/mdev = 0.564/4.973/16.870/6.880 ms
```

3. Si la connexion est fonctionnelle, vous verrez des réponses de ping.

## 2. Étude préliminaire : Solutions de supervision

### Solutions étudiées

#### 1. Nagios

- **Fonctionnalités** : Supervision avancée, notifications par email/SMS, extensibilité via plugins.
- **Prérequis** : Installation manuelle, configuration complexe.
- **Facilité d'installation** : Modérée, nécessite une bonne compréhension des fichiers de configuration.
- **Cas d'usage** : Supervision de réseaux complexes avec besoins personnalisés.

#### 2. Zabbix

- **Fonctionnalités** : Tableau de bord interactif, alertes en temps réel, prise en charge des agents sans configuration.
- **Prérequis** : Serveur avec MySQL/PostgreSQL, installation modérée.
- **Facilité d'installation** : Bonne, documentation claire et interface conviviale.
- **Cas d'usage** : Suivi des performances pour moyennes et grandes entreprises.

#### 3. Centreon

- **Fonctionnalités** : Interface web avancée, gestion centralisée, reporting complet.
- **Prérequis** : Serveur Linux (RedHat/CentOS recommandé), MySQL/MariaDB.
- **Facilité d'installation** : Facile grâce à l'assistant intégré.
- **Cas d'usage** : Surveillance étendue avec un accent sur les tableaux de bord et les rapports.

### Choix de la solution

Nous avons choisi **Zabbix** pour les raisons suivantes :

- **Interface intuitive** : Convient pour une configuration rapide et une prise en main simple.
- **Open source** : Gratuit et bien documenté.
- **Adaptabilité** : Idéal pour superviser des environnements de taille moyenne avec des alertes personnalisées.

## 3. Installation et configuration du logiciel

### Installation de Zabbix sur Cyclop OS (Gestionnaire)

#### Étape 1 : Préparation de l'environnement

Mettez à jour le système :

```
sudo apt update && sudo apt upgrade
```

Installez les prérequis nécessaires :

```
sudo apt install apache2 mysql-server php php-mysql
```

#### Étape 2 : Installation de Zabbix

Téléchargez le dépôt officiel :

```
wget https://repo.zabbix.com/zabbix/6.4/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest_6.4+ubuntu22.04_all.deb
sudo dpkg -i zabbix-release_latest_6.4+ubuntu22.04_all.deb
sudo apt update
```

1. Installez Zabbix Server, le frontend, et l'agent :

```
sudo apt install zabbix-server-mysql zabbix-frontend-php zabbix-agent
zabbix-sql-scripts zabbix-apache-conf
```

#### Étape 3 : Configuration de la base de données

Connectez-vous à MySQL pour créer une base de données :

```
sudo mysql
```

#### Réinitialiser le mot de passe root MySQL (si nécessaire)

Si vous avez besoin de définir ou de réinitialiser le mot de passe root pour MySQL, procédez comme suit :

Arrêtez le service MySQL :

```
sudo systemctl stop mysql
```

Lancez MySQL en mode sans échec :

```
sudo mysqld_safe --skip-grant-tables &
```

Connectez-vous à MySQL sans mot de passe :

```
sudo systemctl start mysql  
mysql -u root
```

Une fois connecté, exécutez les commandes suivantes pour mettre à jour le mot de passe root :

```
USE mysql;  
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'iampassword';  
FLUSH PRIVILEGES;
```

Redémarrez le service MySQL :

```
sudo systemctl start mysql
```

Essayez de vous reconnecter avec le nouveau mot de passe :

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

Dans MySQL, exécutez les commandes suivantes :

```
CREATE DATABASE zabbix CHARACTER SET utf8mb4 COLLATE utf8mb4_bin;  
CREATE USER 'zabbix'@'localhost' IDENTIFIED BY 'zabbixpassword';  
GRANT ALL PRIVILEGES ON zabbix.* TO 'zabbix'@'localhost';  
FLUSH PRIVILEGES;  
  
set global log_bin_trust_function_creators = 1;
```

Importez les données initiales :

```
zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql -u zabbix -p  
zabbix
```

**Vérifier l'importation** Après l'exécution de la commande, connectez-vous à MySQL pour vérifier que les tables ont été créées :

```
mysql -u zabbix -p  
set global log_bin_trust_function_creators = 0;
```

Une fois connecté, utilisez la base de données Zabbix et affichez les tables :

```
USE zabbix;  
SHOW TABLES;
```

```
mysql> use zabbix;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
mysql> show tables;
```

Tables_in_zabbix
acknowledges
actions
alerts
auditlog
autoreg_host
changelog
conditions
config
config_autoreg_tls
connector
connector_tag
corr_condition
corr_condition_group
corr_condition_tag
corr_condition_tagpair
corr_condition_tagvalue
corr_operation
correlation
dashboard

## Étape 4 : Lancement de Zabbix

- Configurez le fichier `/etc/zabbix/zabbix_server.conf` pour ajouter les informations de la base de données.

On met les paramètres de la base de données :

```
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=zabbixpassword
```

Redémarrez les services :

```
sudo systemctl restart zabbix-server zabbix-agent apache2
```

## Configuration de l'agent Zabbix sur Kali Linux (Agent)

Installez l'agent Zabbix :

```
wget
https://repo.zabbix.com/zabbix/6.4/debian/pool/main/z/zabbix-release/zabbix-relea
se_latest_6.4+debian12_all.deb
dpkg -i zabbix-release_latest_6.4+debian12_all.deb

sudo apt update
sudo apt install zabbix-agent
```



1. Modifiez le fichier de configuration `/etc/zabbix/zabbix_agentd.conf`

```
Server=192.168.56.101
ServerActive=192.168.56.101
Hostname=KaliLinux
```

2. Redémarrez l'agent :

```
sudo systemctl restart zabbix-agent
```

Pour s'assurer que le hostname est bien résolu, il faudrait vérifier le hostname de la machine :

```
sudo nano /etc/hostname
```

ou on met cette valeur :

```
KaliLinux
```

```
sudo nano /etc/hosts
```

ou on met cette valeur :

```
127.0.1.1    KaliLinux
```

Et ensuite on update :

```
sudo systemctl restart systemd-hostnamed
```

On vérifie les logs pour voir que tout se passe bien :

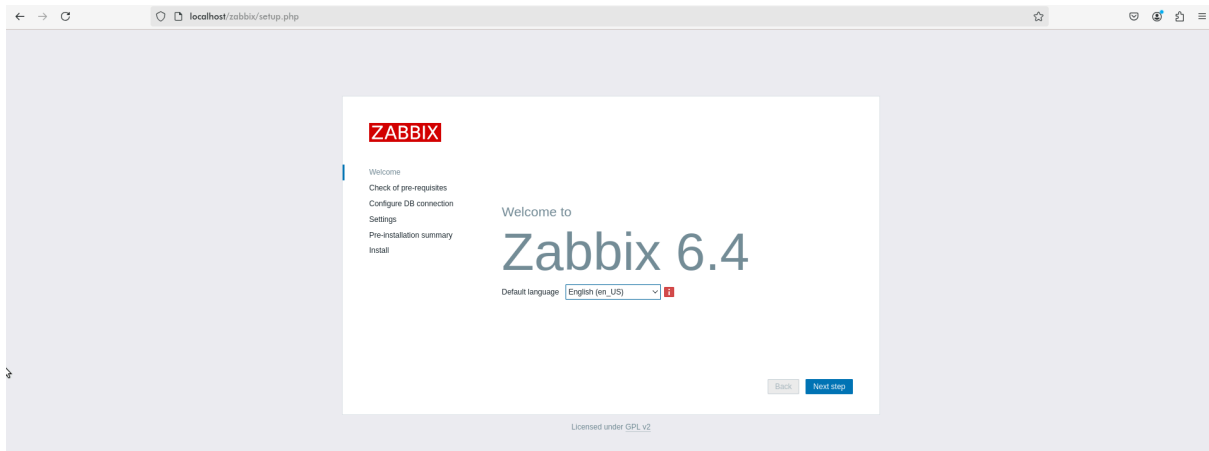
```
(rahmonex@vbox)-[~]
$ sudo tail -f /var/log/zabbix/zabbix_agentd.log
73818:20250128:103833.274 IPv6 support:      YES
73818:20250128:103833.274 TLS support:      YES
73818:20250128:103833.274 *****
73818:20250128:103833.274 using configuration file: /etc/zabbix/zabbix_agentd.conf
73818:20250128:103833.275 agent #0 started [main process]
73819:20250128:103833.276 agent #1 started [collector]
73820:20250128:103833.277 agent #2 started [listener #1]
73821:20250128:103833.277 agent #3 started [listener #2]
73823:20250128:103833.280 agent #5 started [active checks #1]
73822:20250128:103833.283 agent #4 started [listener #3]
```

## 4. Test et validation

1. Configuration des hôtes dans Zabbix :
  - Ajoutez un nouvel hôte depuis l'interface web.

- Configurez les éléments à surveiller (service Apache).

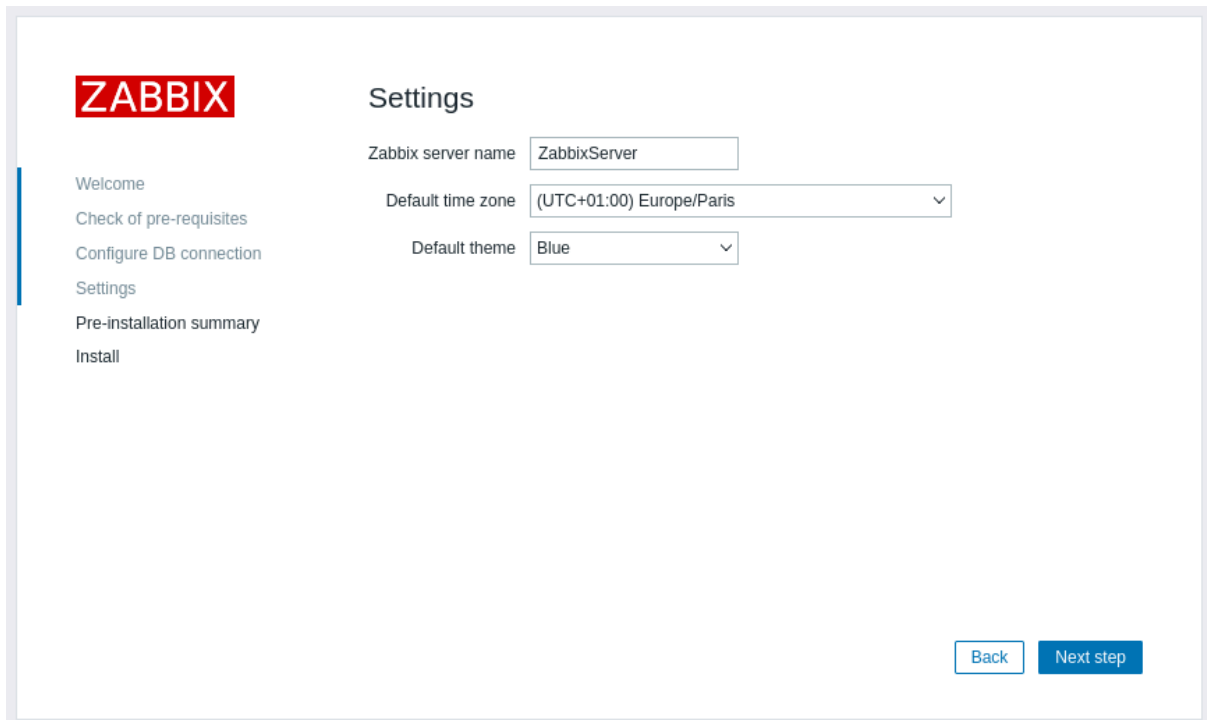
On ouvre ensuite l'interface graphique de Zabbix en allant sur <http://localhost/zabbix> :



On configure ensuite la base de données :

A screenshot of the 'Configure DB connection' step in the Zabbix installation process. The page features the Zabbix logo on the left and a sidebar with navigation links: Welcome, Check of pre-requisites, Configure DB connection (active), Settings, Pre-installation summary, and Install. The main content area is titled 'Configure DB connection' and includes instructions: 'Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.' The configuration fields include: 'Database type' (a dropdown menu set to 'MySQL'), 'Database host' (a text input field with 'localhost'), 'Database port' (a text input field with '0' and a note '0 - use default port'), 'Database name' (a text input field with 'zabbix'), 'Store credentials in' (three tabs: 'Plain text' is selected, followed by 'HashiCorp Vault' and 'CyberArk Vault'), 'User' (a text input field with 'zabbix'), and 'Password' (a masked text input field with dots). At the bottom, there is a 'Database TLS encryption' section with a note: 'Connection will not be encrypted because it uses a socket file (on Unix) or shared memory (Windows).' At the bottom right, there are 'Back' and 'Next step' buttons.

Ensuite on met le bon fuseau horaire, et on définit un nom de serveur :

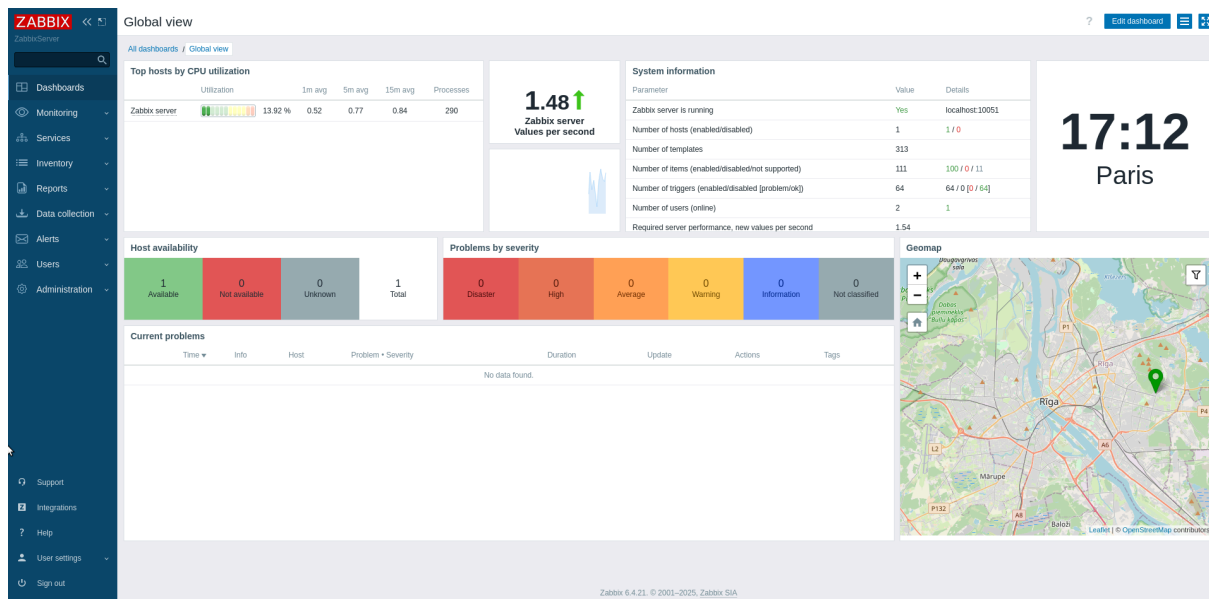
The image shows the Zabbix web interface during installation. On the left is a sidebar with the ZABBIX logo and a list of steps: Welcome, Check of pre-requisites, Configure DB connection, Settings (highlighted), Pre-installation summary, and Install. The main area is titled 'Settings' and contains three configuration fields: 'Zabbix server name' with the value 'ZabbixServer', 'Default time zone' with a dropdown menu showing '(UTC+01:00) Europe/Paris', and 'Default theme' with a dropdown menu showing 'Blue'. At the bottom right are two buttons: 'Back' and 'Next step'.

]

Ensuite on se connecte avec les identifiants par défaut :

- *Utilisateur : Admin*
- *Mot de passe : zabbix*

Et on se connecte au dashboard :



## 2. Simulation de problèmes :

Arrêtez le service Apache sur Kali Linux :

```
sudo systemctl stop apache2
```

Vérifiez que Zabbix génère une alerte.

## 3. Analyse réseau avec Wireshark :

- Capturez le trafic réseau entre 192.168.56.101 et 192.168.56.102.
- Identifiez les protocoles utilisés, comme SNMP ou HTTP.

## 5. Monitoring de services

On crée tout d'abord le hôte dans la page monitoring > Hosts, et on appuie sur create host :

**Host**

Host IPMI Tags Macros Inventory Encryption Value mapping

\* Host name

Visible name

Templates

\* Host groups

Interfaces

Type	IP address	DNS name	Connect to	Port	Default
Agent	<input type="text" value="192.168.56.102"/>	<input type="text"/>	<input checked="" type="radio"/> IP <input type="radio"/> DNS	<input type="text" value="10050"/>	<input checked="" type="radio"/> <a href="#">Remove</a>

[Add](#)

Description

Monitored by proxy

Enabled ☒