

Mission 14 : Supervision de l'infrastructure

0. Pré-requis

Un **serveur web** apache2 avec php, le module PDO et quelques modules php nécessaires dans une nouvelle VM nommée zabbix1-srv sur **le VLAN SRV**

1. Introduction

Le nombre de machines et d'équipements réseau augmentent, il devient vitale de mettre en place **un service de supervision/monitoring**. Pour cela nous utiliserons une solution **OpenSource** largement utilisée dans le milieu professionnel, **Zabbix**.



Zabbix permet de suivre en temps réel l'état du parc informatique : charge CPU, mémoire, espace disque, disponibilité des services, trafic réseau, etc. Il peut également envoyer des alertes (mail, Discord, etc.) lorsqu'un problème est détecté.

La supervision repose principalement sur deux méthodes :

- **Les agents Zabbix (agent2)** installés sur les serveurs, qui remontent des informations détaillées (CPU, RAM, services, logs...)
- **Le protocole SNMP**, utilisé pour superviser les équipements réseau tels que les switchs ou routeurs



Contrairement à de simples outils de vérification de disponibilité, Zabbix stocke toutes les données dans une **base MySQL** et les présente via une **interface web** (Apache2 + PHP), ce qui permet de surveiller l'infrastructure, d'analyser l'historique et de créer des tableaux de bord personnalisés.

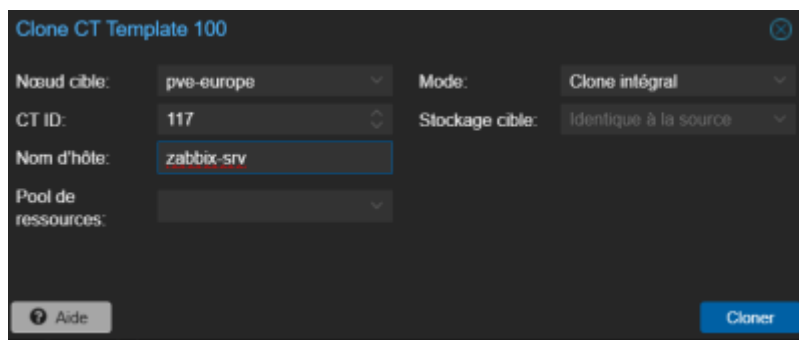
Avec Zabbix, il devient possible :

- De surveiller l'ensemble des serveurs (SRV, DMZ...),
- De superviser les switchs via SNMP,
- De créer des triggers personnalisés,
- D'envoyer des alertes automatiques en cas de problème.

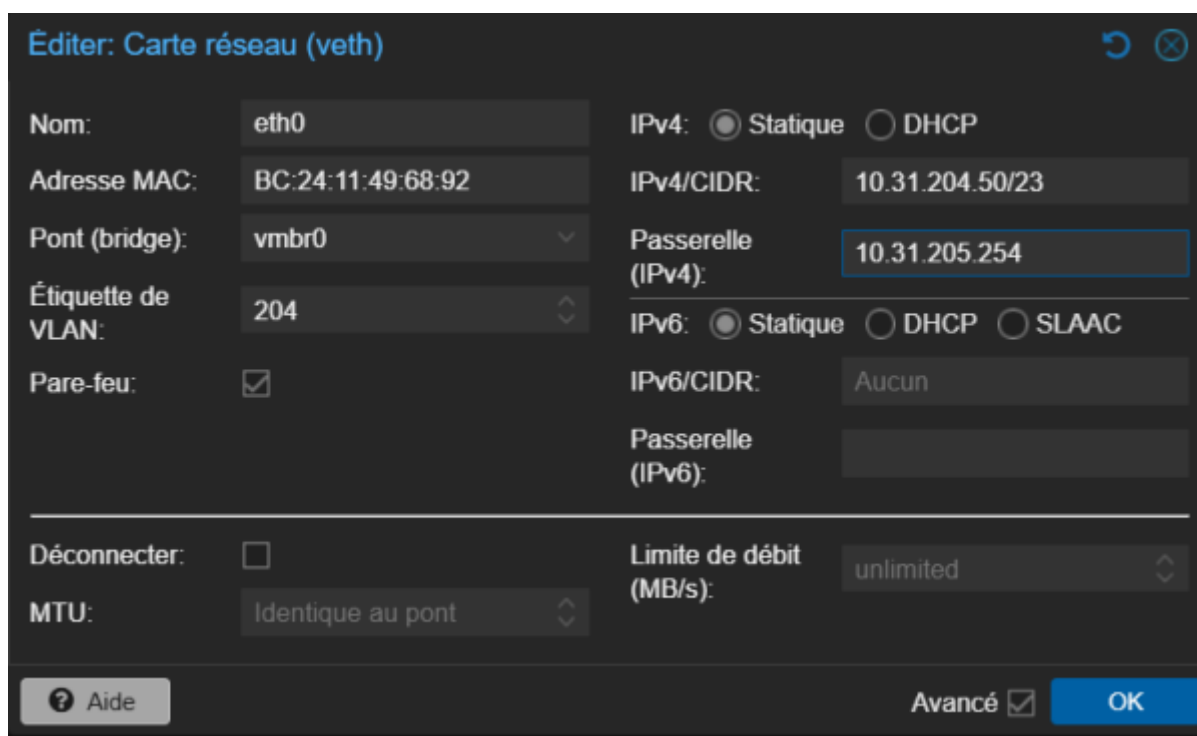
2. Création et configuration du serveur zabbix

2.1 Création du conteneur zabbix1-srv

On fais un clone du conteneur **template** :



On modifie directement sa **configuration réseau** (en oubliant pas **l'étiquette du VLAN SRV** :



On met les paquets à jour :

```
root@zabbix1-srv:~# apt update
root@zabbix1-srv:~# apt upgrade
```

Comme demandé, on installe **apache2 avec php, les modules pdo, et quelques modules php**. Cela nous permettra d'accéder à zabbix sur l'interface **web** :

```
root@zabbix1-srv:~# apt install apache2 php php-mysql php-mysqldb php-ldap
php-bcmath php-mbstring php-gd php-pdo php-xml libapache2-mod-php
```

On se rend sur le officiel de zabbix pour installer le logiciel : <https://www.zabbix.com/fr/download>

On effectue la configuration par rapport à notre **système** et notre **utilisation** :

- Nous installerons la **version 7.4** de zabbix

VERSION DE ZABBIX	OS DISTRIBUTION	VERSION DU SYSTÈME D'EXPLOITATION	ZABBIX COMPONENT	BASE DE DONNÉES	SERVEUR WEB
7.4	Alma Linux	13 Trixie (amd64, arm64)	Server, Frontend, Agent	MySQL	Apache
7.2	Amazon Linux	12 Bookworm (amd64, arm64)	Server, Frontend, Agent 2	PostgreSQL	Nginx
7.0 LTS	CentOS	11 Bullseye (amd64, arm64)	Proxy		
6.0 LTS	Debian	10 Buster (amd64, i386)	Agent		
8.0 PRE-RELEASE	OpenSUSE Leap		Agent 2		
	Oracle Linux		Java Gateway		
	Raspberry Pi OS		Web Service		
	Red Hat Enterprise Linux				
	Rocky Linux				
	SUSE Linux Enterprise Server				
	Ubuntu				

Notes de publication 7.4

Ensuite, on suit la **procédure d'installation de zabbix**.

1. Installer le **répertoire Zabbix** :

```
root@zabbix1-srv:~# wget
https://repo.zabbix.com/zabbix/7.4/release/debian/pool/main/z/zabbix-
release/zabbix-release_latest_7.4+debian13_all.deb #récupere le fichier
depuis l'URL

root@zabbix1-srv:~# dpkg -i zabbix-release_latest_7.4+debian13_all.deb
#ajout du référentiel zabbix dans notre système (pour faire apt install)
root@zabbix1-srv:~# apt update #met à jour les paquets
```

2.Installation de **zabbix server, frontend, agent2** :

```
root@zabbix1-srv:~# apt install zabbix-server-mysql zabbix-frontend-php
zabbix-apache-conf zabbix-sql-scripts zabbix-agent2
```

3. Installation des plugins zabbix agent 2 :

```
root@zabbix1-srv:~# apt install zabbix-agent2-plugin-mongodb zabbix-agent2-
plugin-mssql zabbix-agent2-plugin-postgresql
```

4. Création de la **base de donnée**

On se rend dans notre **conteneur priv-db1** pour **créer la base de donnée et l'utilisateur zabbix** qui nous servira à **stocker les données** pour superviser nos conteneurs.

```
root@priv-db1:~# mysql -uroot -p #connexion à MariaDB
Mariadb> create database zabbix character set utf8mb4 collate utf8mb4_bin;
```

```
Mariadb> create user 'zabbix'@'10.31.204.50' identified by 'password';  
Mariadb> grant all privileges on zabbix.* to 'zabbix'@'10.31.204.50';  
Mariadb> set global log_bin_trust_function_creators = 1;
```

5. Import du schéma de **base de donnée**

On retourne dans la machine **zabbix1-srv** pour **décompresser tous les scripts SQL nécessaires à la base Zabbix** et se **connecter à l'utilisateur zabbix** à distance depuis notre serveur zabbix :

On a eu un petit problème...

Dans le **doc de la mission** (#sabotage du professeur), il nous ait demandé de taper la commande suivante :

```
root@zabbix1-srv:~# : zcat /usr/share/zabbix-sql-  
scripts/mysql/server.sql.gz | mysql --default-characterset=utf8mb4 -h  
10.31.X.33 -uzabbix -ppassword zabbix
```

Sauf que la commande n'a pas fonctionné car **le chemin du fichier server.sql.gz** dans notre système linux debian13 **n'est pas le même** que celui renseigné.

Donc on a fait :

```
root@zabbix1-srv:~# find /usr/share -name "server.sql.gz"  
résultat affiché : /usr/share/zabbix/sql-scripts/mysql/server.sql.gz  
#chemin du fichier server.sql.gz
```

Maintenant qu'on a le **véritable chemin du fichier**, on adapte notre commande.

Note importante :

: la commande **zcat** qui crée la base de données à partir du **script sql est prévue pour un serveur mysql sur la même machine que le serveur zabbix**.

Ce n'est pas notre cas, nous devons l'adapter : `zcat /usr/share/zabbix/sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -h 10.31.204.33 -uzabbix -ppassword zabbix`

On tape donc la commande dans le serveur **zabbix1-srv** :

```
root@zabbix1-srv:~# zcat /usr/share/zabbix/sql-scripts/mysql/server.sql.gz |  
mysql --default-character-set=utf8mb4 -h 10.31.204.33 -uzabbix -ppassword  
zabbix
```

Attention :

On a l'impression qu'il y a un bug mais **la commande** met un certain temps à s'exécuter donc ne surtout pas mettre **ctrl+d** pour **l'interrompre**

6. Désactiver l'option temporaire (c'est une faille potentielle)

On se rend de nouveau dans **notre base de donnée priv-db1** et on tape :

```
root@priv-db1:~# mysql -uroot -p
Mariadb> set global log_bin_trust_function_creators = 0;
Mariadb> quit;
```

7. Configuration de la bdd pour le serveur zabbix

On va permettre au service Zabbix à se connecter à la base en allant dans le fichier de conf du serveur :

```
zabbix1-srv:~# nano /etc/zabbix/zabbix_server.conf
```

Et on cherche la ligne "DBPassword=" et on met :

```
/etc/zabbix/zabbix_server.conf
DBPassword=password
```

C'est le mot de passe que **Zabbix** utiliseras pour se **connecter à sa base de donnée**.

8. Démarrer les services :

```
root@zabbix1-srv:~# systemctl restart zabbix-server zabbix-agent2 apache2
root@zabbix1-srv:~# systemctl enable zabbix-server zabbix-agent2 apache2
```

9. Installer les locales en_US.UTF8 :

```
root@zabbix1-srv:~# apt install locales
root@zabbix1-srv:~# dpkg-reconfigure locales
```

On sélectionne **en_US.UTF8** On redémarre **apache**, on **refresh** .

10. On accède à l'interface web de Zabbix :

On accède à zabbix via l'url : <http://10.31.204.50/zabbix>

3. Configuration sur l'interface

3.1 Première configuration de l'interface

1. On vérifie que les **modules PHP** nécessaires sont installés, **sinon on corrige** :

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Check of pre-requisites

	Current value	Required	
PHP version	7.4.33	7.4.0	OK
PHP option "memory_limit"	128M	128M	OK
PHP option "post_max_size"	16M	16M	OK
PHP option "upload_max_filesize"	2M	2M	OK
PHP option "max_execution_time"	300	300	OK
PHP option "max_input_time"	300	300	OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK
PHP option "mbstring.func_overload"	off	off	OK

Back

Next step

2. On configure la **connexion du serveur Zabbix** à la **Base de donnée** :

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database.
Press "Next step" button when done.

Database type

MySQL

Database host

10.31.204.33

Database port

3306

0 - use default port

Database name

zabbix

Store credentials in

Plain text

HashiCorp Vault

CyberArk Vault

User

zabbix

Password

Database TLS encryption

☐

Back

Next step

3. On renseigne le **nom du serveur**, la **timezone** et le **thème par défaut** :

```
Zabbix server name : zabbix1-srv
Default time zone : (UTC+01:00) Europe/Paris
Default theme : Dark
```

4. On **vérifie** puis **lance** l'installation et on **se loggue sur Zabbix** :

Nom d'utilisateur : **Admin**

Mot de passe : **zabbix**

3.2 Configuration de l'utilisateur et des vm host

1. Création d'un utilisateur :

Dans l'interface **zabbix**, on se rend dans **users** → **users** → **create user**

On peut, à partir de là, créer un **utilisateur**, lui assigner un **groupe** et lui donner les **permissions** souhaitées.

Voici l'**utilisateur** que nous avons **créé** :

☐	Adem	Adem	Lazri	Super admin role	Zabbix administrators	No
--------------------------	------	------	-------	------------------	-----------------------	----

2. Création des host :

On se rend dans **Monitoring** → **Host** → **Create host**

Exemple de configuration pour la machine Backup1 (10.31.204.73) :

The screenshot shows the Zabbix 'Add host' configuration page. The 'Host name' is 'backup-01' and the 'Visible name' is also 'backup-01'. Under 'Templates', 'Linux by Zabbix agent' is selected. In the 'Host groups' section, 'Linux servers' and 'Zabbix servers' are selected. The 'Interfaces' table shows one interface with 'Type' as 'Agent', 'IP address' as '10.31.204.73', and 'Port' as '10050'. The 'Description' field is empty. At the bottom, 'Monitored by' is set to 'Server' and the 'Enabled' checkbox is checked.

□ A quoi correspondent les infos pour crée un host ?

- **Host name** : nom unique de la machine.
- **Templates** : modèles de supervision (ex : "Linux by Zabbix agent" pour CPU, RAM, disque...).
- **Host groups** : groupes logiques pour organiser les machines (ex : Linux servers, Zabbix servers).
- **Interfaces** :
 1. **Type** : Agent, SNMP...
 2. **IP address / DNS name** : adresse du host
 3. **Port** : 10050 par défaut pour Zabbix Agent

3. Configuration des host :

Après avoir **créé et configuré un host** sur le **serveur Zabbix**, on se rend dans la **configuration** de cette même **machine** pour configurer **l'agent**, exemple pour backup1 :

```
root@backup01-srv:~# apt update
root@backup01-srv:~# apt upgrade
root@backup01-srv:~# apt install zabbix-agent2
root@backup01-srv:~# nano /etc/zabbix/zabbix_agent2.conf
```

On change la ligne : **server**=127.0.0.1 en **server**=10.31.204.50

```
root@backup01-srv:~# systemctl restart zabbix-agent2
```

On fais la **même chose** pour toutes nos **machines**.

3.3 Supervision du switch

Désormais, nous allons configurer notre **switch** pour pouvoir le **supervisé**.

□ Supervision d'un switch (version courte)

□ Rappel express : SNMP sur un switch

- **SNMP** : méthode utilisée pour superviser les équipements réseau.
- **Port utilisé** : UDP **161** (requêtes SNMP).
- **Community** : mot-clé d'accès, ex : *public* (lecture seule).
- **Interface dans Zabbix** : choisir **SNMP**, mettre l'IP du switch.
- **Template** : utiliser un modèle **Switch SNMP** pour récupérer trafic + état des ports automatiquement.

□ **En bref** : pas d'agent sur un switch → Zabbix interroge directement l'appareil via SNMP.

1. Configuration du switch :

On se connecte au **switch** physiquement via un **cable console cisco**.

L'objectif est de créer une **adresse ip** pour le switch sur l'interface 206 (**VLAN MANAGEMENT**) pour pouvoir le **superviser**.

On configure l'interface VLAN et l'adresse IP de management :

```
Switch> enable      # mode privilégié
Switch> conf t      # mode configuration globale

Switch(config)> interface vlan 206      # VLAN de management
Switch(config-if)> ip address 10.31.206.50 255.255.254.0  # IP du switch
Switch(config-if)> ip default-gateway 10.31.239.254      # passerelle par
défaut
Switch(config-if)> no shut                # active
l'interface
Switch(config-if)> exit
```

□ À quoi ça sert ?

- Donner une **adresse de management** au switch,
- Permettre à Zabbix et aux autres équipements d'y accéder,
- La **passerelle** est nécessaire pour communiquer avec les autres VLANs (ex : VLAN SRV où se trouve Zabbix).

On configure SNMP en supprimant la communauté par défaut et en créant une communauté

sécurisée :

```
Switch# conf t          # entrer en mode config

Switch(config)> no snmp-server community public R0      # supprime la
communauté par défaut
Switch(config)> snmp-server community fls R0            # crée une communauté
SNMP (lecture seule)
Switch(config)> snmp-server host 10.31.204.50 fls       # autorise Zabbix à
interroger le switch
Switch(config)> end                                         # quitter le mode
config
Switch> write memory                                       # sauvegarde la
configuration
```

❏ Pourquoi SNMP ?

- Zabbix utilise **SNMP** pour lire l'état du switch,
- Port utilisé : **UDP 161**,
- La communauté **fls** sert de clé d'accès,
- Permet de superviser : trafic, ports, erreurs, uptime, etc.

2. Création et configuration de l'host sur Zabbix :

Ayant configuré le **switch**, il nous faut maintenant **créer** et **configurer** correctement le **host** pour **superviser** le switch.

Voici la **configuration** à **effectuer** :

The screenshot shows the Zabbix 'Host' configuration window. The 'Host' tab is selected. The 'Host name' field is set to 'switch-mgt'. The 'Visible name' field is also 'switch-mgt'. Under 'Templates', 'Cisco IOS by SNMP' is selected. Under 'Host groups', 'Linux servers' and 'Zabbix servers' are selected. In the 'Interfaces' section, an interface of type 'SNMP' is configured with IP address '10.31.204.50', connecting to 'IP' on 'DNS' port '161'. The 'Default' button is highlighted. The 'Description' field is empty. At the bottom, 'Monitored by' is set to 'Server', and the 'Enabled' checkbox is checked. The 'Update' button is highlighted.

□ Explication des paramètres du host SNMP (switch)

- **Host name : switch-mgt**

→ Nom utilisé par Zabbix pour identifier le switch.

- **Template : Cisco IOS by SNMP**

→ Modèle prévu pour les switches/routeurs Cisco.

→ Ajoute automatiquement : état des ports, trafic, erreurs, CPU, uptime...

- **Host groups :**

→ Classement du switch dans un ou plusieurs groupes.

→ Permet une meilleure organisation (ex : "Zabbix servers" ou "Network devices").

- **Interface SNMP :**

1. **Type : SNMP** → Zabbix interroge le switch via SNMP, pas via un agent.
2. **IP address : 10.31.206.50** → Adresse de management configurée sur le VLAN mgt.
3. **Port : 161** → Port standard SNMP (UDP).
4. **Community** → Définie côté switch (ex : *fls*), et renseignée dans le template.

□ **En bref :** Ce host est configuré pour que Zabbix supervise le switch via SNMP sur le port 161, avec le template Cisco prévu pour analyser automatiquement ses interfaces et son état général.

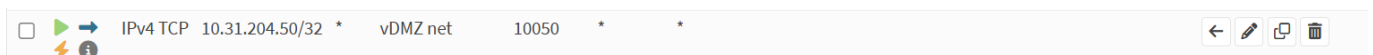
3.4 Règles de Parefeu

Pour que nos actions puissent **fonctionner**, il faut que l'on crée des règles de **parefeu**. Ces règles **autoriserons** les requêtes **nécessaires** au fonctionnement de nos **services**.

1. Règle autorisant la supervision des machines du réseau DMZ

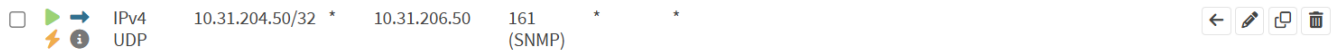
Toute nos machines du **réseau DMZ** ont besoins d'être **supervisées**.

On fais donc une **règle** qui **autorise** notre conteneur **Zabbix (10.31.204.50)** à faire des **requêtes** sur **le port 10050** (port utilisé par l'agent zabbix) vers le **réseau DMZ** :



2. Règle autorisant la supervision du switch

On fais une règle qui autorise notre conteneur **Zabbix (10.31.204.50)** à faire des requêtes SNMP sur le port 161 à l'IP de notre **switch (10.31.206.50)** :



4. Configuration de l'alerte discord

4.1 Création de l'alerte Discord

Pour mettre en place une alerte discord cela s'effectue en plusieurs étapes : Tout d'abord nous allons ajouter la macro `{ZABBIX.URL}` qui va

From:

<https://sisr2.beaupeyrat.com/> - Documentations SIO2 option SISR

Permanent link:

<https://sisr2.beaupeyrat.com/doku.php?id=sisr2-europe:zabbix>

Last update: **2026/03/13 14:41**

