Mise en place d'un serveur de supervision Zabbix

Sommaire

Cahier des charges – Expression des besoins

Descriptif de l'existant

L'infrastructure de ma ferme de serveurs est déjà en place : les deux pares-feux pfSense en haute disponibilité (CARP / pfSync), les contrôleurs de domaine Active Directory (WS2025-AD01 et WS2025-AD02) avec les services DNS, le serveur MariaDB (DEB13-MariaDB), le serveur GLPI (DEB13-GLPI), le serveur Nextcloud (DEB13-Nextcloud), le reverse proxy (DEB13-ReverseProxy), les serveurs web (DEB13-Web-01 et DEB13-Web-02) ainsi que les postes clients et le poste d'administration. Seul le système de sauvegarde n'est pas encore en place, mais cela n'a pas d'impact sur le déploiement de Zabbix. L'ensemble des réseaux (CLIENTS, SERVEURS, ADMINS, DMZ-FRONT, DMZ-BACK) est segmenté et routé à travers les pfSense, avec un accès Internet fonctionnel.

Besoin(s)

Je dois mettre en place un serveur de supervision sur ma ferme de serveurs afin de pouvoir surveiller l'état de l'ensemble de mes machines (serveurs Debian 13, Windows Server 2025, postes clients, équipements réseau). L'objectif est d'avoir une visibilité en temps réel sur la disponibilité des serveurs et des services, et de détecter rapidement si un serveur est en panne ou si un service est à l'arrêt. Je n'ai pas mis en place de remontées d'alertes (e-mail, SMS, etc…) car la ferme de serveurs est arrêtée chaque fois que j'ai fini de travailler dessus, ce qui rendrait les alertes inutiles dans mon contexte.

Contrainte(s)

L'installation et la configuration de Zabbix doivent être terminées avant le 13 mai 2026. De plus, les ressources de la ferme n'étant pas illimitées, j'ai limité la VM DEB13-Zabbix à 1 processeur, 2 Go de RAM et 30 Go de disque.

Ressources

Ressources mises à disposition

J'ai à ma disposition un serveur Proxmox VE hébergeant l'ensemble de ma ferme de serveurs virtuelle, ainsi qu'un accès Internet via mes 2 pares-feux pfSense. J'ai également à ma disposition l'ISO de Debian 13.

Ressources nécessaires à la mise en place

Pour réaliser cette mise en place, j'ai besoin de l'ISO de Debian 13, des paquets nécessaires à l'installation de Zabbix Server 7.4 (serveur, frontend web Apache, agent Zabbix 2) et d'une connexion Internet via mes pares-feux pfSense pour télécharger les dépôts officiels Zabbix. La base de données utilisée par Zabbix est hébergée sur le serveur DEB13-MariaDB (172.16.2.12), déjà en place sur le réseau SERVEURS.

Gestion des ressources

Je dispose déjà de l'ISO Debian 13. J'ai dû créer la VM DEB13-Zabbix de zéro sur Proxmox, la VM a comme ressources : 1 vCPU, 2 Go de RAM, 30 Go de disque et 2 cartes réseau (une pour le réseau SERVEURS et l'autre pour le réseau ADMIN). L'accès Internet nécessaire au téléchargement des paquets est assuré par le routage existant via les 2 pfSense.

Analyse

Descriptifs des solutions

La supervision d'une infrastructure peut être assurée par différents outils. Je vais comparer deux solutions open-source de référence : Centreon et Zabbix.

Centreon : plateforme de supervision basée historiquement sur Nagios, proposant une interface web complète et un système de plugins pour la collecte de métriques. Centreon existe en version open-source (Centreon IT Edition) et en version commerciale avec des fonctionnalités avancées (cartographie, reporting, HA intégrée).

Zabbix : solution de supervision open-source intégrée (serveur, agents, frontend web, base de données) permettant la collecte de données via agents, SNMP, IPMI ou JMX. Zabbix propose nativement des templates préconfigurés, des tableaux de bord personnalisables et une gestion fine des alertes. L'ensemble des fonctionnalités est disponible gratuitement sans restriction.

Comparaison des solutions (tableau)

Critères	Centreon	Zabbix
Coûts	Version open-source gratuite. Version commerciale (IT-100, Business, MSP) payante pour les fonctionnalités avancées (cartographie, reporting, HA).	Entièrement gratuit et open-source (licence GPL v2). Toutes les fonctionnalités sont accessibles sans restriction. Support commercial optionnel.
Administration	Interface web complète. Configuration centrée sur les plugins et les modèles d'hôtes. Gestion des droits intégrée.	Interface web complète (frontend PHP). Configuration via templates, gestion granulaire des droits utilisateurs.
Intégration SI	Bonne intégration avec les environnements hétérogènes via plugins. Connecteurs LDAP / AD disponibles.	Très bonne intégration : agents natifs Linux / Windows, support SNMP, IPMI, JMX. Authentification LDAP / AD native. API REST complète.
Protocoles de collecte	Principalement SNMP et plugins Nagios. Agents Centreon disponibles.	Agents Zabbix natifs (actif / passif), SNMP (v1 / v2c /v3), IPMI, JMX, HTTP / HTTPS, scripts personnalisés.

Templates / Modèles	Modèles d'hôtes (Plugin Packs), certains payants en version commerciale.	Large bibliothèque de templates officiels et communautaires, tous gratuits. Templates préconfigurés pour Linux, Windows, réseau, etc.
Tableaux de bord	Tableaux de bord personnalisables. Cartographie (MAP) en version commerciale.	Tableaux de bord entièrement personnalisables, graphiques, cartes réseau, le tout inclus dans la version gratuite.
Journalisation	Logs applicatifs, historique des événements, rapports (version commerciale pour les rapports avancés).	Historique complet des données collectées, journal d'audit, rapports intégrés. Conservation configurable.
Déploiement	Installation via script ou ISO dédiée (CentOS / AlmaLinux). Configuration initiale guidée.	Installation via paquets officiels (Debian, Ubuntu, RHEL, etc.). Configuration initiale via le frontend web.
Maintenance	Mises à jour via gestionnaire de paquets. Procédure documentée.	Mises à jour via gestionnaire de paquets depuis les dépôts officiels Zabbix. Procédure documentée.
Courbe d'apprentissage	Moyenne à élevée (concepts Nagios hérités, plugins, configuration des modèles).	Moyenne. La documentation officielle est très complète et la communauté active facilite la prise en main.

Choix d'une solution – Argumentation

Je choisis de mettre en place Zabbix, car il est entièrement gratuit et open-source sans restriction de fonctionnalités, contrairement à Centreon dont certaines fonctionnalités (cartographie, reporting avancé, haute disponibilité) sont réservées aux versions commerciales. Zabbix propose nativement des agents multiplateformes (Linux et Windows), une authentification LDAP / AD intégrée, une large bibliothèque de templates gratuits et une API REST complète. La documentation officielle est très complète et la communauté est active, ce qui facilite la prise en main et le dépannage. Enfin, l'installation via les paquets officiels sur Debian est simple et bien documentée, ce qui correspond à mon environnement.

Plan d'adressage - Schéma - Tables de routage

- Tableau d'adressage :

Nom du réseau	@ réseau	Masque	1er @ IP	Dernière @ IP	@ broadcast	Passerelle	Nombre d'hôtes
WAN	10.0.0.0	255.255.0.0	10.0.0.1	10.0.255.254	10.0.255.255	10.0.205.251	65536
SERVEURS	172.16.2.0	255.255.255.0	172.16.2.1	172.16.2.254	172.16.2.255	172.16.2.1	254
ADMINS	172.16.3.0	255.255.255.0	172.16.3.1	172.16.3.254	172.16.3.255	172.16.3.1	254

- Tableau des IPs :

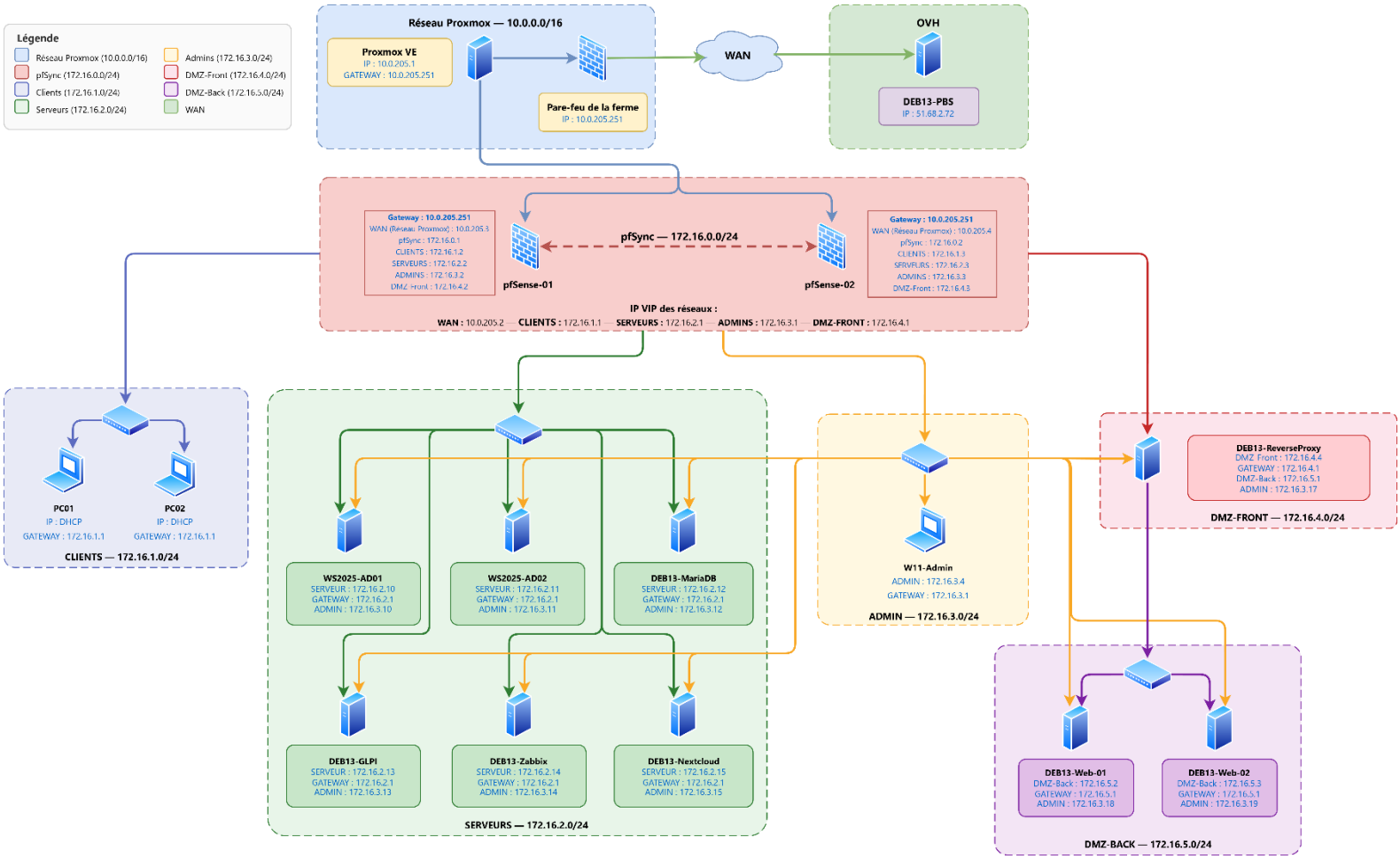
Réseau	VIP	pfSense-01	pfSense-02	W11-Admin	WS2025-AD01	DEB13-MariaDB	DEB13-Zabbix
WAN	10.0.205.2	10.0.205.3	10.0.205.4				
SERVEURS	172.16.2.1	172.16.2.2	172.16.2.3		172.16.2.10	172.16.2.12	172.16.2.14
ADMINS	172.16.3.1	172.16.3.2	172.16.3.3	172.16.3.4	172.16.3.10	172.16.3.12	172.16.3.14

- Table de routage des routeurs pfSense :

PfSense-01			
Adresse réseau	**Masque réseau**	**Adresse passerelle**	**Interface**
0.0.0.0	0.0.0.0	10.0.205.251	10.0.205.2
10.0.205.0	255.255.0.0	10.0.205.2	10.0.205.2
172.16.2.0	255.255.255.0	172.16.2.2	172.16.2.2
172.16.3.0	255.255.255.0	172.16.3.2	172.16.3.2

PfSense-02			
Adresse réseau	**Masque réseau**	**Adresse passerelle**	**Interface**
0.0.0.0	0.0.0.0	10.0.205.251	10.0.205.3
10.0.205.0	255.255.0.0	10.0.205.3	10.0.205.3
172.16.2.0	255.255.255.0	172.16.2.3	172.16.2.3
172.16.3.0	255.255.255.0	172.16.3.3	172.16.3.3

- Schéma du réseau :



Étude de l'impact sur le SI existant

L'intégration d'un serveur de supervision Zabbix dans l'infrastructure apporte une visibilité complète sur l'état des machines et des services. Elle constitue une brique essentielle pour la détection d'incidents et le suivi de la disponibilité, mais impacte plusieurs aspects du système d'information.

- **Impact technique :**

La mise en place de Zabbix nécessite une VM dédiée (DEB13-Zabbix) hébergeant le serveur Zabbix, le frontend web Apache et l'agent Zabbix 2. La base de données est déportée sur le serveur DEB13-MariaDB (172.16.2.12) déjà en place, ce qui introduit une dépendance réseau entre les deux machines : si MariaDB est indisponible, Zabbix ne peut plus stocker ni consulter les données de supervision. L'agent Zabbix 2 doit être déployé sur l'ensemble des serveurs (serveurs Debian 13, Windows Server 2025). L'ensemble de la communication entre le serveur Zabbix et les agents s'effectue via le réseau ADMINS (172.16.3.0/24), sur lequel toutes les machines disposent déjà d'une interface, ce qui évite d'avoir à modifier les règles de pare-feu sur les pfSense. Le trafic généré par la collecte des métriques reste faible en fonctionnement normal, mais la rétention des données historiques peut faire croître significativement la taille de la base de données si la durée de conservation n'est pas

correctement configurée. Enfin, les templates utilisés doivent être adaptés à chaque type d'hôte pour éviter des vérifications inutiles ou des faux positifs.

- **Impact organisationnel :**

Zabbix introduit une supervision centralisée qui modifie les pratiques d'exploitation. Les techniciens n'ont plus besoin de vérifier manuellement l'état de chaque machine : les tableaux de bord permettent de visualiser l'ensemble de l'infrastructure en temps réel. En contrepartie, les templates et les seuils d'alerte doivent être maintenus à jour lors de l'ajout de nouvelles machines ou de nouveaux services. Chaque ajout d'hôte doit suivre une procédure documentée (installation de l'agent, déclaration dans Zabbix, affectation du bon template, vérification de la collecte). Dans un contexte de production, la configuration des alertes (e-mail, SMS) devrait également être formalisée pour définir qui reçoit quelle alerte et selon quel niveau de criticité.

- **Impact stratégique :**

La supervision renforce la capacité de détection et de réaction face aux incidents. Un serveur en panne ou un service arrêté est identifié immédiatement au lieu d'être découvert tardivement par un utilisateur. Cela améliore la continuité de service et permet une exploitation plus proactive de l'infrastructure. Les données historiques collectées par Zabbix permettent également d'identifier des tendances (saturation disque, augmentation de charge, dégradation de performances) et d'anticiper les incidents avant qu'ils ne surviennent. Cette visibilité facilite aussi la prise de décision lors de l'ajout de ressources ou de la planification de maintenances.

- **Impact humain :**

Les techniciens doivent monter en compétence sur le fonctionnement de Zabbix : configuration des templates, compréhension des items et des triggers, personnalisation des tableaux de bord, et diagnostic des problèmes de collecte (agent non joignable, item non supporté, erreur de template). La navigation dans l'interface web de Zabbix demande un temps d'adaptation, notamment pour exploiter efficacement les graphiques, les cartes réseau et les journaux d'événements. En exploitation, l'équipe doit être capable de distinguer rapidement un vrai incident d'un faux positif lié à un seuil mal configuré.

- **Impact juridique :**

Zabbix collecte des données exclusivement techniques sur les serveurs supervisés : nom d'hôte, adresse IP, charge CPU, utilisation mémoire, espace disque, état des services, temps de disponibilité, etc. Ces données ne constituent pas des données personnelles puisqu'elles concernent des serveurs et non des postes utilisateurs. Néanmoins, les journaux de supervision doivent être conservés de manière sécurisée, avec une durée de rétention définie et un accès restreint aux administrateurs. Ces traces peuvent être mobilisées lors d'audits internes ou d'investigations pour reconstituer l'état de l'infrastructure à un instant donné.

Phasage de l'intervention

Je commencerai par créer la VM DEB13-Zabbix sur Proxmox VE et installer Debian 13. Ensuite, j'installerai le serveur Zabbix 7.4 avec le frontend Apache et l'agent Zabbix 2 depuis les dépôts officiels, puis je créerai la base de données sur le serveur DEB13-MariaDB et j'importerai le schéma SQL initial. Je mettrai ensuite en place HTTPS sur le frontend web via un certificat signé par mon autorité de certification AD CS, puis je terminerai la configuration initiale de Zabbix via l'assistant web. Je configurerai ensuite l'authentification LDAP pour permettre la connexion avec les comptes Active Directory. Une fois le serveur opérationnel, je déploierai l'agent Zabbix 2 sur l'ensemble des hôtes à superviser, en commençant par les serveurs Debian 13 puis les serveurs Windows Server 2025, et j'ajouterai chaque hôte dans l'interface Zabbix avec le template adapté. Enfin, je vérifierai la collecte des données.

Prévision des tests de validation

Après l'installation du serveur Zabbix, je vérifierai que les services **zabbix-server**, **zabbix-agent2** et **apache2** sont bien démarrés et fonctionnels. Je testerai l'accès au frontend web en HTTPS et je vérifierai que le certificat AD CS est bien reconnu par le navigateur. Après la configuration de LDAP, je testerai l'authentification en me connectant à l'interface Zabbix avec un compte Active Directory. Après le déploiement de l'agent Zabbix 2 sur les hôtes, je vérifierai que l'agent est bien démarré et que les hôtes apparaissent bien comme disponible (icône verte) dans l'interface Zabbix. Je contrôlerai également que la collecte de données fonctionne en vérifiant la remontée des métriques (CPU, RAM, disque, réseau) dans les dernières données de chaque hôte. Enfin, je testerai le bon fonctionnement des déclencheurs en arrêtant volontairement un serveur supervisé et en vérifiant que Zabbix détecte bien le problème et l'affiche dans le tableau de bord.

Mise en place

La partie Mise en place a été rédigée sur Notion, elle est accessible via le lien suivant : https://mathys-demon.notion.site/Mise-en-place-d-un-serveur-Zabbix-33e8102ecdc98008bf39dc04f674de08.

Bilan

Conclusion

Le serveur de supervision Zabbix a été mis en place avec succès en environ deux soirées. L'ensemble des serveurs de la ferme est désormais supervisé et la collecte de données fonctionne comme prévu. Les tableaux de bord permettent de visualiser l'état de l'infrastructure en temps réel.

Auto-évaluation

J’ai réussi à installer et configurer Zabbix 7.4 sur Debian 13 avec Apache, en utilisant la base de données MariaDB hébergée sur le serveur DEB13-MariaDB, à déployer l’agent Zabbix 2 sur l’ensemble des hôtes de ma ferme de serveurs et à vérifier que la collecte de données fonctionnait correctement. La partie la plus formatrice a été la compréhension du fonctionnement des templates, qui permettent de standardiser la supervision sans avoir à configurer chaque élément manuellement. J’ai également appris à naviguer dans l’interface web de Zabbix et à personnaliser les tableaux de bord. Si c’était à refaire, je prendrais le temps de configurer également les alertes, afin de maîtriser cette fonctionnalité pour un futur déploiement en production.