

Avenue du ciseau 15,

1348 Ottignies-Louvain-la-Neuve

**Implémentation d’un système de surveillance et d’alerte externe d’une infrastructure réseau modulaire**

Travail de fin d’étude réalisé en vue de l’obtention du diplôme de bachelier en Informatique et Systèmes orientation Technologie de l’informatique

**Joel Cédric YEPGANG NAMENYI**



Rapporteur : **Marie-Noël VROMAN**

**Année Académique 2018-2019**

# Avant -propos

La troisième année du cycle bachelier en Informatique et Systèmes est marquée au second semestre par l’élaboration d’un travail de fin d’études (TFE). Ce dernier a principalement pour objectif d’initier, de former et de familiariser l’étudiant que je suis à la réalisation d’un travail de recherche conciliant à la fois une étude théorique et un cas pratique à présenter devant un jury constitué d’enseignants et des professionnels. L’objet de ce travail de fin d’études, sa présentation et sa défense devraient permettre de prouver ma capacité à pouvoir mener à bien un projet autonome et personnel et donc à m’intégrer dans un milieu socio-professionnel.

Cette unité d’enseignement (TFE) correspond à 16 ECTS et donc à une période de travail estimée à 480 heures.

# Remerciements

Avant tout développement de ce rapport,il apparait judicieux de remercier ceux sans qui ce travail n’aurait pas été possible.

J’adresse ainsi mes remerciements à tous ceux qui ont eu la gentillesse de faire de ce travail de fin d’études un moment très intéressant. Il s’agit très spécifiquement de Mme Marie-Noël VROMAN mon rapporteur pour ses conseils, la patience et le professionnalisme pédagogique donc elle a fait preuve tout au long de nos multiples entretiens. Ainsi que, Mr Laurent Schalkwijk pour les échanges et les réponses à mes questions durant la réalisation de ce projet.

Je remercie également l’ensemble du corps professoral du Département Technologie de l’informatique de l’EPHEC pour le travail abattu durant ce cursus.

Enfin, je remercie également, ma famille, ma compagne et mes amis qui m’ont beaucoup soutenu moralement, physiquement et ont été toujours là depuis le début.

[Avant -propos 2](#_Toc17153734)

[Remerciements 3](#_Toc17153735)

[Introduction 6](#_Toc17153736)

[I. Cahier de charges 1](#_Toc17153737)

[II. Etude théorique du projet 2](#_Toc17153738)

[II.1 Importance du projet 2](#_Toc17153739)

[II.2 Difficultés liées au projet 2](#_Toc17153740)

[II.3 Études comparatives des outils de surveillance 2](#_Toc17153741)

[II.3.1 Les outils payants 2](#_Toc17153742)

[II.3.2 Les outils open source 3](#_Toc17153743)

[III. Environnement de travail 4](#_Toc17153744)

[III.1 Les OS (operating système) 4](#_Toc17153745)

[III.2 Les logiciels 4](#_Toc17153746)

[IV. Architecture du système d’information 4](#_Toc17153747)

[IV.1 Architecture existante 4](#_Toc17153748)

[IV.2 Architecture définitive 5](#_Toc17153749)

[IV.2.1 Description 6](#_Toc17153750)

[V. Mise en place de l’ossature du système de contrôle 6](#_Toc17153751)

[V.1 Système de simulation 6](#_Toc17153752)

[V.1.1 GNS3 6](#_Toc17153753)

[V.1.1.1 Définition 6](#_Toc17153754)

[V.1.1.2 Raison de choix 6](#_Toc17153755)

[V.1.2 VMWARE VM (Virtual machine) Workstation pro 7](#_Toc17153756)

[V.1.2.1 Définition 7](#_Toc17153757)

[V.1.2.2 Avantages de cet hyperviseur 7](#_Toc17153758)

[V.2 Procédure d’intégration des hôtes 8](#_Toc17153759)

[V.2.1 Description 8](#_Toc17153760)

[V.2.2 Conclusion 9](#_Toc17153761)

[VI. Système d’intégration de l’architecture réseau 10](#_Toc17153762)

[VI.1.1 Schéma 10](#_Toc17153763)

[VI.1.2 Mise en place des technologies 11](#_Toc17153764)

[VI.1.2.1 Pfsense 11](#_Toc17153765)

[VI.1.2.1.1 Fonctionnalités 12](#_Toc17153766)

[VI.1.2.1.2 Avantages 12](#_Toc17153767)

[VI.1.2.2 Les vpn 13](#_Toc17153768)

[VI.1.2.2.1 Protocol IPSec 13](#_Toc17153769)

[VI.1.2.2.2 Protocol OpenVPN 14](#_Toc17153770)

[VI.1.2.2.2.1 Serveur 15](#_Toc17153771)

[VI.1.2.2.2.2 Client 15](#_Toc17153772)

[VI.1.3 Développement du système de contrôle 16](#_Toc17153773)

[VI.1.3.1 Zabbix 16](#_Toc17153774)

[VI.1.3.1.1 Zabbix serveur 17](#_Toc17153775)

[VI.1.3.1.2 Zabbix-agent 17](#_Toc17153776)

[VI.1.3.1.3 Zabbix-front-end 17](#_Toc17153777)

[VI.1.3.1.4 Zabbix proxy 17](#_Toc17153778)

[VI.1.3.2 Intégration de la base de données partagées 17](#_Toc17153779)

[VI.1.3.3 Haute disponibilité 18](#_Toc17153780)

[VI.1.3.4 Pacemaker 18](#_Toc17153781)

[VI.1.3.5 Corosync 19](#_Toc17153782)

[VII. Systèmes de monitoring 19](#_Toc17153783)

[VII.1.1.1 Description 19](#_Toc17153784)

[VII.1.1.1.1 Intégration d’un hôte à monitorer 20](#_Toc17153785)

[VII.1.1.1.2 Intégration d’un système d’alerte 20](#_Toc17153786)

[VIII. Difficultés rencontrées 21](#_Toc17153787)

[IX. Améliorations possibles 21](#_Toc17153788)

[X. Recommandations 21](#_Toc17153789)

[XI. Conclusion 22](#_Toc17153790)

[Bibliographie 23](#_Toc17153791)

# Introduction

Dans le cadre de notre parcours à l’Ephec, la dernière année est sanctionnée par le développement d’un sujet représentant notre projet de fin d’études. La finalité étant de mener à bien une mission venant d’un besoin exprimé ou non par une entreprise. Dans cette optique, l’élaboration de celui-ci nous permettra lors de nos multiples recherches d’acquérir une expérience socio-professionnelle puisqu’il intègre de la rigueur, de la méthode et de l’organisation. Le but visé par ce projet est de démontrer qu’avec nos acquis d’apprentissage accumulés durant notre cursus, on est capable non seulement de comprendre un problème concret, c’est-à-dire mener une analyse, rechercher des solutions et d’en réaliser des cas pratiques, mais aussi de prouver sa capacité d’insertion socioprofessionnelle par une réalisation autonome.

Ainsi, lors d’une discussion avec un ami, il m’a soulevé une problématique à laquelle il fait face chaque fois dans son réseau. Je le citerai mot pour mot : « chaque fois que je suis à l’extérieur, je ne suis prévenu d’aucune manière de l’arrêt de mes systèmes dans mon infrastructure et même je n’ai aucun suivi de ce qui s’est passé en mon absence. Et en plus si un système tombe, je devrais tout réinstaller à nouveau. ». Ainsi se posait le problème. À savoir : comment garder en permanence un œil sur son réseau tout en réduisant le temps d’intervention sur celui-ci lors des défaillances systèmes ? Dans ce cadre, il s’agit ici d’une :

**Implémentation d’un système de surveillance et d’alerte externe d’une infrastructure réseau modulaire**

Etant définie comme Théorie et traitement de l'information à l'aide de programmes mis en œuvre sur ordinateurs, l’informatique est en constante évolution et par conséquent très complexe quel que soit son domaine d’activité. Le monitoring (surveillance) vu ici comme composant de contrôle est une activité importante que doit incorporer dans ses services une entreprise hébergeant un parc informatique car, la grande partie des entreprises fonctionnent aujourd’hui à travers le réseau. Ainsi la disponibilité et la qualité de services conditionnent le bon fonctionnement de l’entreprise. Dès lors, comment mettre en place de manière sécurisée, un système de surveillance qui permettra de détecter rapidement certains processus défaillants ?   
Dans le souci de mieux gérer et contrôler l’activité de l’infrastructure, ce système non seulement garantira la disponibilité des équipements du parc informatique en cas de panne ou baisse de performance, mais aussi tentera de faire une prévention des problèmes pouvant survenir. Il sera donc question pour nous d’être informé ou d’être notifié en temps réel des possibles cas de problèmes, d’activité interrompue et de l’état de notre réseau qui peut être dû à une coupure d’électricité, coupure d’internet etc.

Tout en présentant le cahier de charges, ce rapport fera ressortir dans un premier temps une étude théorique du sujet, par la suite présentera l’architecture du système, puis décrira au travers de cette architecture le système de collecte et stockage des données qui seront par la suite analysées aux fins de créations des différentes actions(notifications).

# Cahier de charges

Bien que celui-ci soit réalisé dans un environnement de simulation, cette rubrique présente les directives qu’on essayera d’atteindre.

Partant du fait que, un client(virtuel) d’une entreprise(virtuelle) désire une solution de surveillance de son réseau pour prévoir des éventuels problèmes, Ci-dessous sont reprises les différentes étapes par lesquelles sont passées ce projet.

Système de monitoring

Un serveur externe au réseau (plus script pour installation et config) qui servira à l’installation du moteur de contrôle. Dans la situation envisagée ce dernier sera en failover avec un autre serveur situé à l’intérieure du réseau.

Un serveur local sous Debian (plus script pour installation et config) qui servira à l’installation du moteur de contrôle et en failover avec le serveur externe.

Pour des raisons de sécurité, un serveur différent contiendra une base de données partagée aux deux serveurs en failover. Ainsi on conserve les informations en cas de basculement.

Appareils à monitorer

Une **VM(VMWare) Debian** fonctionnel plus un script .sh pour tout réinstaller servira de machine à monitorer à cela s’ajoute un Backup journalier vers le NAS

* Apache PHP 7.2, SSL (Let's Encrypt);
* MySQL Server ;
* OpenLDAP ;
* SSH ;
* Zabbix-agent.

Les pfsenses qui permettront non seulement la connexion des deux sites distants, mais aussi permettront les connexions de l’extérieur au réseau local avec un backup journalier vers le NAS. En outre les éléments à surveiller seront :

* Utilisation cpu/ram, hdd
* Statistique réseau : trafic en sur chaque interface (+/- temps réel, trafic coulé sur la journée, la semaine, le mois, total) plus éventuellement par protocoles
* Logs pare-feu (20 derniers au moins)
* Etat des services, des interfaces
* Clients connectés sur les LANs, VPN et leurs IP/MAC .

Une machine Windows avec un agent Windows installé pour récupérer les informations sûres : l’utilisation cpu/ram, hdd.

Switch (Cisco SG500-28P) avec agent SNMP pour surveiller son statut et l’état des ports

Nas (Synology et Qnap)

* Etats (On/Off) Utilisation cpu/ram, hdd
* Température du cpu, hdd
* Utilisateurs connectés Logs (20 derniers au moins)
* Etats de services
* Date et heures du dernier backup Infos backup (réussis, raté > log)

# Etude théorique du projet

Afin d’avoir une idée claire lors de la réalisation de ce projet, il est important d’étudier la faisabilité de celui-ci à travers des thèmes existant en rapport avec celui-ci. Ainsi cette rubrique présentera l’importance de ce projet, étudiera les difficultés liées à celui-ci, pour en finir avec une étude comparative des différents moteurs de contrôle existant sur le marché.

Importance du projet

Au regard de ce sujet, il est surtout important dans la mesure où il intègre plusieurs notions acquises durant ma formation et certaines de mes multiples formations réalisées en ligne. Il s’agit entre autres de :

* **Gestion de projet*:*** c’est une démarche visant à organiser de bout en bout le bon déroulement d'un projet. Une bonne gestion des facteurs opérationnels et de méthodologie font qu'un projet puisse aboutir dans un triangle représentant l'équilibre qualité-coût-délai (QCD).
* **Sécurité informatique**: discipline qui vise la protection de l'intégrité et la confidentialité des informations stockées dans un système informatique, en outre le contrôle des utilisateurs et des droits d’accès aux données.
* **Service réseau** : fonctionnalité assurée par un ou plusieurs serveurs ayant pour fonction la fourniture d'informations à d'autres ordinateurs via une connexion réseau normalisée.
* **SGBD** : système qui stocke et gère des données de façon organisée et cohérente.
* **SQL**: langage permettant l’interrogation des données stockées dans les SGBD.

Difficultés liées au projet

En ce qui concerne les difficultés de ce projet, elles vont des thèmes principaux qui sont monitoring et alerte qui ont été jusqu’ici nouveaux pour nous. Bien qu’abordés en parallèle durant le cursus à travers le mot SNMP, ceci restait des expressions auxquels, il fallait trouver un sens enfin qu’ils puissent correspondre au sujet. Et face à ceci, il nous fallut au préalable étudié les différentes solutions de surveillance présente sur le marché.

Études comparatives des outils de surveillance

Comme mentionné plus haut, il existe donc plusieurs systèmes de surveillance donc ici nous les classerons en deux catégories à savoir : les outils payants et les outils open source .

### Les outils payants

On entend par outils payants un logiciel développé et commercialisé par une ou un groupe de personne. Ayant un peu parcouru la documentation, on y trouvera :

OP5 E-Monitor et PRTG qui sont des logiciels destinés aux serveurs. La surveillance et la gestion du réseau. L’utilisation de ce dernier est limitée à un essai de 30 jours après quoi, il nous faudra une licence.

### Les outils open source

On entend par open source un logiciel gratuit qu’on peut non seulement télécharger et installer gratuitement, mais aussi configurer selon ses propres besoins. Il en existe plusieurs dont ce rapport présente ceux ayant lu la documentation. Il s’agit de :

* Zabbix
* Nagios

Table comparative des outils suscités

|  |  |  |  |
| --- | --- | --- | --- |
| **Catégories** | **Outils** | **Avantages** | **Inconvénients** |
| **Open source** | Zabbix  Nagios | Gratuit, facilité d’installation pour les débutants | Documentation et interface graphique complexe  Non réactivité du développeur sur le partage des modules |
| **Payante** | PRTG  OP5 | Offre un essaie de son environnement pour 30 en limité  Installation très facile | Logiciel à but commercial |

# Environnement de travail

Bien qu’ayant une entreprise virtuelle, tout produit se doit d’être développé aux fins d’utilisation. Et, pour sa facilité d’installation et de configuration, ce projet est élaboré sur des environnements et logiciels bien connus avec un accès facile aux différents serveurs par des clients appropriés.

Les OS (operating système)

* Os Windows utilisés pour l’installation des logiciels de simulation réseau.
* Os Debian ou Ubuntu pour l’installation des services et logiciels de contrôle.

Les logiciels

* GNS3 ;
* VMWare ;
* Edraw max.

# Architecture du système d’information

Architecture existante

Comme tout travail, il convient d’analyser l’existant avant toute modification possible.

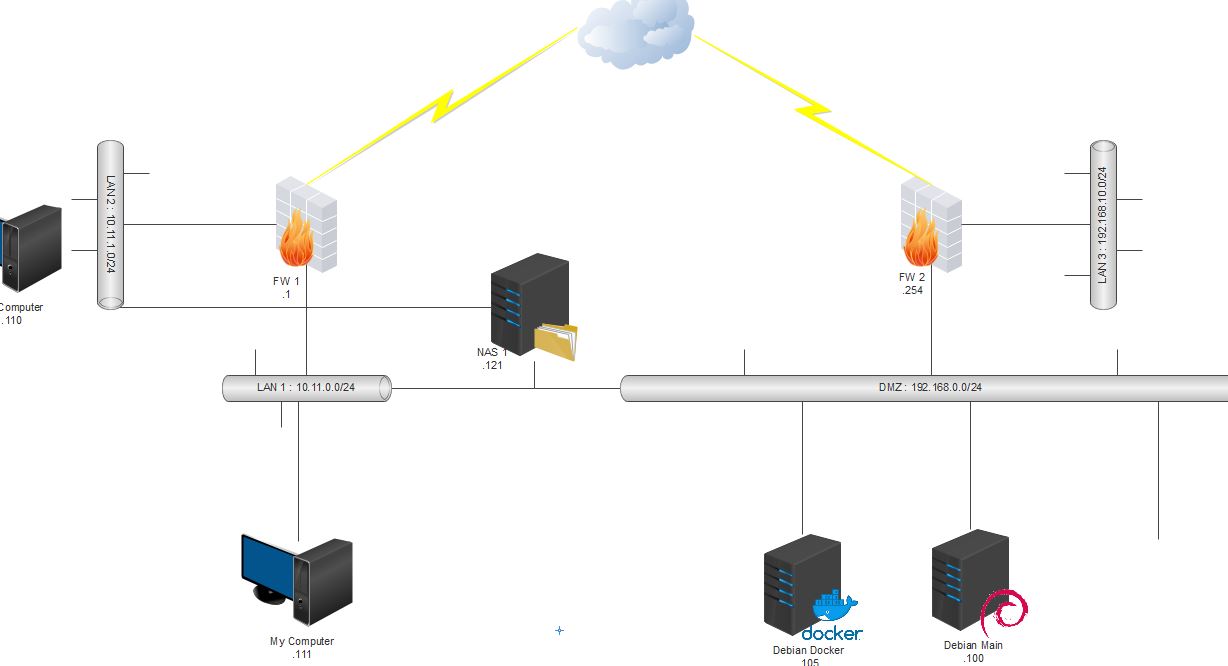


Fig.1 réseau existant

Le réseau ainsi défini, correspond à l’architecture initiale ou de base qui a permis d’élaborer une réflexion sur ce travail. Il est principalement constitué de  :

* Trois LAN : LAN1, LAN2 , LAN3
* Une DMZ qui héberge les machines du réseau interne qui ont besoin d’être accessibles depuis l’extérieur
* Deux firewalls donc un sur le site principale FW1 et un autre sur le site distant FW2
* Un NAS pour la sauvegarde journalière des données.

Afin de le rendre évolutif et mieux adapté aux standings d’un réseau moderne, il a fallu procéder à une analyse et une mise à niveau correspondant aux attentes du cahier de charge et aux perspectives d’évolution qui permettraient de mieux contrôler les incidents et les interventions potentiels. Ainsi après révision de l’ensemble de réseau, il a fallu envisager des possibles accessibilités externes et internes qui permettraient de maintenir le réseau en activité permanente.

Architecture définitive

Avant de détailler toutes les étapes de réalisation de ce travail, il est important de donner un aperçu du schéma global. À partir de celui-ci chaque étape de réalisation pourra être mieux visualisé et pourra dès lors montrer son utilité et le but de son intégration. A ceci sera joint une explication du choix de chaque technologie.

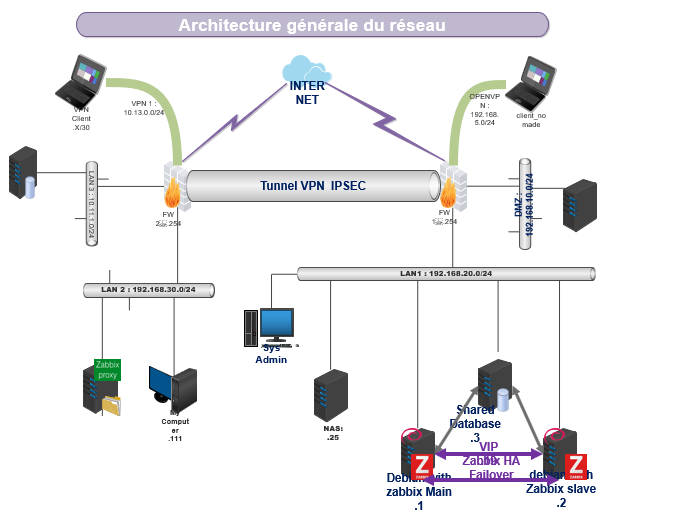


Fig.2 réseau modifier

### Description

En accord avec le cahier de charge, on observe bien au travers de ce schéma des éventuels ajouts qui peuvent se remarquer avec les termes comme OpenVPN qui permettra la connexion par exemple d’un client nomade, tunnel IPSec qui permettra la connexion des deux sites distants, Zabbix pour l’installation du système de contrôle. À noter que tous ces termes seront développés plus bas dans la mise en place des systèmes, services et solution du système de notifications.

# Mise en place de l’ossature du système de contrôle

En rapport avec le cahier de charge ce système doit pouvoir surveiller tous les équipement réseau se trouvant dans parc informatique d’une entreprise. Pour ce faire ce système sera divisé en plusieurs sous-systèmes à savoir : la mise place du système de simulation réseau, l’intégration des différents équipements réseaux nécessaires dans ce système, installation et configuration au niveau des équipements des différents technologies et services en accord avec le cahier de charges, mise en place du système de notification.

Système de simulation

Comme dit plus haut, il est choisi ici comme logiciels de virtualisation GNS3 et VMWARE.

### GNS3

#### Définition

**Gns3** , pour  Graphical Network Simulator est un [logiciel libre](https://fr.wikipedia.org/wiki/Logiciel_libre) permettant l'émulation ou la [simulation](https://fr.wikipedia.org/wiki/Simulation_r%C3%A9seau) de réseaux informatiques. Publié pour la première fois en 2008, Il permet la combinaison de périphériques virtuels et réels, utilisés pour simuler des réseaux complexes. Il utilise le logiciel d'émulation [Dynamips](https://en.wikipedia.org/wiki/Dynamips) pour simuler [Cisco IOS](https://en.wikipedia.org/wiki/Cisco_IOS) .  Il est utilisé par de nombreuses grandes entreprises, notamment [Exxon](https://en.wikipedia.org/wiki/Exxon) , [Walmart](https://en.wikipedia.org/wiki/Walmart) , [AT & T](https://en.wikipedia.org/wiki/AT%26T) et [NASA](https://en.wikipedia.org/wiki/NASA) , et est également populaire pour la préparation des examens de certification professionnelle en réseau. En 2015, le logiciel avait été téléchargé 11 millions de fois.

#### Raison de choix

* Il utilise des fonctions issues du monde de Linux. Sous Windows ou Mac, ses fonctions peuvent être mal interprétées. Le serveur étant sous Ubuntu, cela permet à GNS3 d'exécuter ces fonctions sans problème.
* Il utilise la puissance des logiciels de VMware Workstation et son rendement est supérieur à la simple émulation.
* Il permet d’effectuer une simulation concrète de son réseau avant un déploiement définitif en environnement réel ou physique.
* Il permet l’installation de périphériques autres que les routeurs Cisco. Notons qu’en fonction de la taille du réseau à mettre en place , cette dernière pourra demander plus de ressource. Son utilisation dans ce projet nécessite que l’on crée une liaison réseau à la VMware Workstation, ce qui permet la sauvegarde de toutes manipulations effectuées sur gns3 vers Workstation.

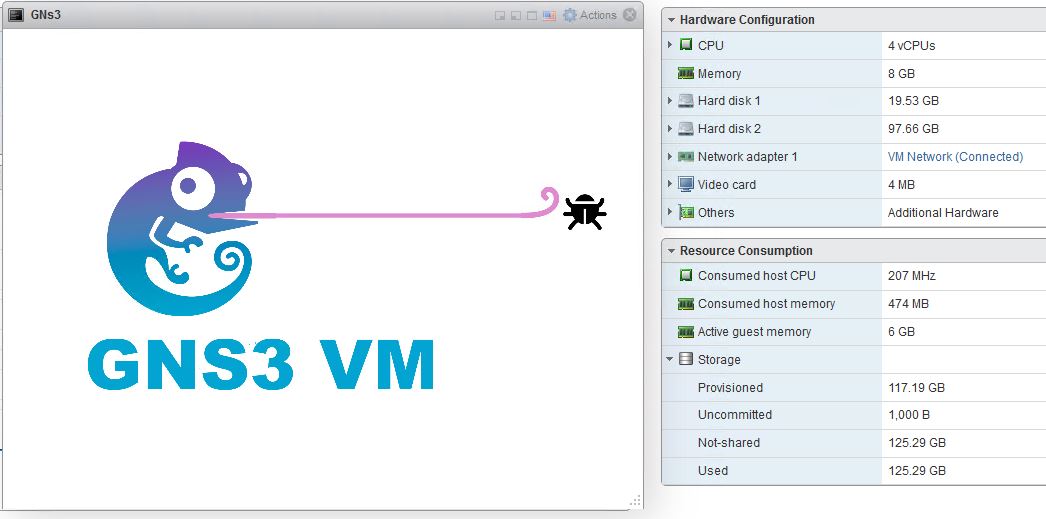


Fig.3 GNS3 VM

### VMWARE VM (Virtual machine) Workstation pro

#### Définition

Ce logiciel est un outil de virtualisation de poste de travail créé par la société [VMware](https://fr.wikipedia.org/wiki/VMware), il peut être utilisé pour mettre en place un environnement de test pour développer de nouveaux logiciels, ou pour tester l'architecture complexe d’un système d’exploitation avant de l’installer réellement sur une machine physique.

Nous avons opté pour une version premium enfin de bénéficier de tout l’environnement qu’offre cette plateforme.

#### Avantages de cet hyperviseur

* **Partitionnement :** Il exécute plusieurs systèmes d’exploitation et repartit les ressources entre les différentes machines virtuelles.
* **Isolation :** Il assure la protection de la sécurité au niveau matériel et les performances accrues tout en déployant le contrôle des ressources.
* **Encapsulation :** Il enregistre des fichiers donnant états des différentes machines virtualisés.

Pour exécuter VMware Workstation, le système hôte et le système d'exploitation hôte doivent satisfaire à des exigences matérielles et logicielles spécifiques. Pour permettre à gns3 d’être hypervisé car celui-ci est gourmand en ressources. Ainsi, Pour pouvoir installer ces logiciels, nous nous sommes procuré une machine faisant office de serveur hôte. Cette **machine serveur**disposera des ressources suivantes

* Système d’exploitation Windows 7 ;
* 64 Go de mémoire RAM (random Access memory) ;
* Processeur Intel Xeon 3.5Ghz ;
* 250 Go de disque dur ssd .

Procédure d’intégration des hôtes

L’installation de gns3 terminée, il nous a été possible par la suite d’importer des images (Appliance) correspondant à un ensemble de périphériques déjà présents dans le système de contrôle de gns3.

Une Appliance sera ici, une machine ou un périphérique déjà présent sur gns3 qui ne peut être utilisé que si l’image correspond à cette dernière est bien installée.

Son importation ne sera possible que depuis le site officiel de gns3.

### Description

Il est important de noter que dans ce système, tous les Appliance utilisées sont gratuits et se trouvent sur la marketplace de GNS3. Il nous sera demandé, de créer un compte utilisateur pour y accéder.

Il nous faudra par la suite, téléchargé et enregistré le système d'exploitation de l’Appliance dans notre répertoire **de travail.**

La figure suivante décrit le processus d’ajout d’une Appliance. Ainsi, nous prendrons en exemple l’ajout d’une machine d’un routeur Cisco sur GNS3.

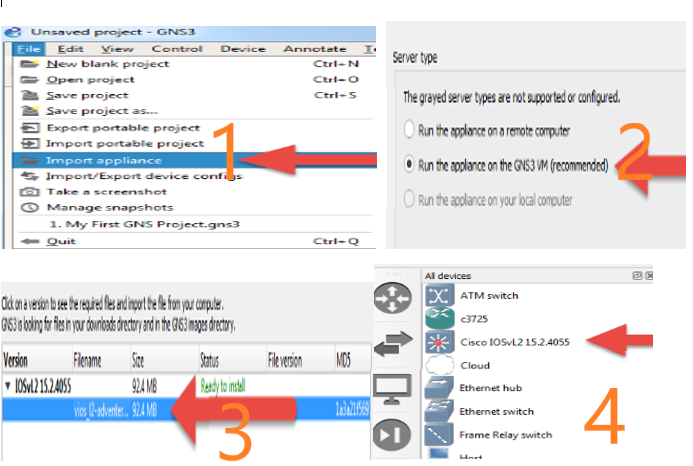


Fig.4 Appliance

### Conclusion

Cette section nous a permis d’intégrer différents os et iOS nécessaire à notre système de contrôle.

# Système d’intégration de l’architecture réseau

Comme , nous pouvons le constater dans la figure reprenant les Appliance installées mentionnée ci-haut, il est important de commencer par configurer les éléments en présence.

La section suivante explique le pourquoi et le choix des périphériques utilisés pour ce projet. Elle reprend dans un ordre bien précis, la démarche entreprise qui a permis d’atteindre la mise en place d’une telle architecture.

### Schéma

Il convient mieux avant d’entrer dans le développement de présenter un schéma logique du réseau pour mieux observer et comprendre l’utilisation de chaque équipement.

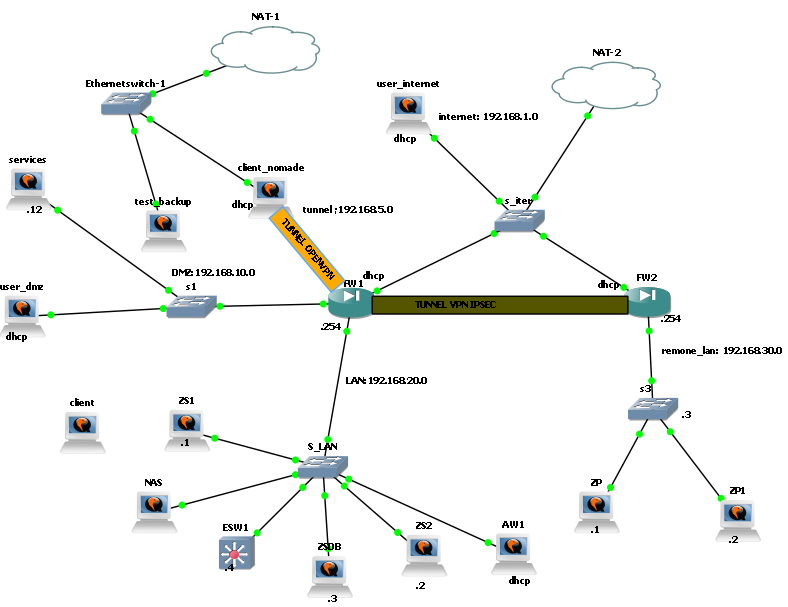


Fig.5 schémas réseau logique.

**FW1**

Sous réseau lan1 IP réseau : 192.168.20.0

Wan (em0) : DHCP

Gateway (em1): 192.168.20.254

IP statique : 192.168.20.1------------------🡪 192.168.30.19

**DMZ**

Sous réseau lan1 IP réseau : 192.168.10.0

Gateway (em1) : 192.168.10.254

Lan optionnel (em2) : 192.168.10.1--------------🡪192.168.10.9

Tunnel OpenVPN : 192.168.5.0

**FW2**

IP réseau : 192.168.30.0

Wan (em0) : DHCP

Lan2 : 192.168.30.254

IP statique : 192.168.30.1------------------🡪 192.168.30.29

### Mise en place des technologies

Comme technologies, nous avons utilisé pour le firewall pfsense enfin de mettre en place comme sécurité les VPN.

#### Pfsense

PfSense est un [routeur](https://fr.wikipedia.org/wiki/Routeur)/[pare-feu](https://fr.wikipedia.org/wiki/Pare-feu_(informatique)) [open source](https://fr.wikipedia.org/wiki/Opensource) basé sur le système d'exploitation [**FreeBSD**](https://fr.wikipedia.org/wiki/FreeBSD). À l'origine il est un [fork](https://fr.wikipedia.org/wiki/Fork_(d%C3%A9veloppement_logiciel)) de [m0n0wall](https://fr.wikipedia.org/wiki/M0n0wall), il utilise le pare-feu à états [Packet Filter](https://fr.wikipedia.org/wiki/Packet_Filter), des fonctions de routage et de [NAT](https://fr.wikipedia.org/wiki/Network_address_translation) lui permettant de connecter plusieurs réseaux informatiques. Il comporte l'équivalent [libre](https://fr.wikipedia.org/wiki/Logiciel_libre) des outils et services utilisés habituellement sur des routeurs professionnels [propriétaires](https://fr.wikipedia.org/wiki/Logiciel_propri%C3%A9taire). PfSense convient pour la sécurisation d'un [réseau domestique](https://fr.wikipedia.org/wiki/R%C3%A9seau_local) ou de petite entreprise.

Après une brève installation manuelle pour assigner les [interfaces réseaux](https://fr.wikipedia.org/wiki/Carte_r%C3%A9seau), il est possible ensuite de l’administrer à distance depuis une [interface web](https://fr.wikipedia.org/wiki/Interface_web). Comme sur les distributions Linux, PfSense intègre aussi un [gestionnaire de paquets](https://fr.wikipedia.org/wiki/Gestionnaire_de_paquets) pour installer des fonctionnalités supplémentaires, comme un proxy, serveur VoIP etc.

##### Fonctionnalités

Il existe une multitude de fonctionnalités liées à pfsense , parmi lesquelles nous pouvons citer les quelques fonctions suivantes :

* Le filtrage

Il permet d’effectuer un filtrage IP source et destination, port du protocole, IP source et destination pour le trafic TCP (Transmission Control Protocol , est un protocole de transport fiable, en [mode connecté](https://fr.wikipedia.org/wiki/Mode_connect%C3%A9) ) et UDP. Ce qui est important lorsque l’on veut limiter les connexions sur une base des règles. cette fonctionnalité de filtrage est activée par défaut sur pfsense et sa désactivation entrainera certains dysfonctionnements NFS (Network File System qui à l'origine est un [protocole](https://fr.wikipedia.org/wiki/Protocole_de_communication)  qui permet à un [ordinateur](https://fr.wikipedia.org/wiki/Ordinateur) d'accéder via un [réseau](https://fr.wikipedia.org/wiki/R%C3%A9seau_informatique) à des [fichiers](https://fr.wikipedia.org/wiki/Fichier_(informatique)) distants).

L’on peut aussi désactiver cette fonctionnalité (filtrage) si l’on souhaite que notre pfsense fonctionne comme un routeur pur ce qui n’est pas le cas dans ce projet.

* Le portail captif

Il permet de forcer l'authentification, ou la redirection vers une page pour l'accès au réseau. Ceci est communément utilisé sur les réseaux de points chauds (Hot Spots) ou encore pour une connexion distante mais est également largement utilisé dans les réseaux d'entreprise pour une couche supplémentaire de sécurité sur l'accès sans fil ou Internet.

* VPN

PfSense offre quatre options de connectivité VPN: IPSec, OpenVPN, PPTP et L2TP.

* Serveur DHCP et relais

PfSense comprend à la fois les fonctionnalités de serveur DHCP et de relais DHCP.

##### Avantages

Une particularité de son utilisation est l’apport technologique lié à sa disponibilité ( load balacing), confidentialité assurée par ces différents protocoles (https authentification, IPSec, OpenVPN etc.), sa simplicité d’installation et d’administration qui requiert néanmoins des connaissances informatiques.

Dans notre cas , l’installation de pfsense est utilisée pour exploiter l’ensemble de ces services . Mais encore grâce à sa fonction vpn, j’ai pu établir une connexion distante entre les deux sites mais aussi établir une connexion sécurisée d’un utilisateur externe.

#### Les vpn

Un vpn est un réseau privé virtuel crypté dans le réseau Internet, qui permet à une société dont les locaux seraient géographiquement dispersés de communiquer et partager des documents.

Il dispose de plusieurs protocoles dont deux ont particulièrement retenu notre attention, il s’agit de IPSec et OpenVPN.

##### Protocol IPSec

C’est un protocole de tunneling qui permet d’établir une connexion entre deux sites distants tout comme OpenVPN mais ce dernier nécessite préalablement l’installation d’un logiciel agent sur le client pour que ce dernier puisque contacter le serveur. A la différence des autres , il permet de véhiculer d’autres protocoles tels que SSH, RDP, SMB, SMTP, IMAP, etc.

Il fallut dans un premier temps fixer les adresses IP au niveau des interfaces Wan des deux firewalls. Ce qui a permis dans notre cas d’activer et d’installer sur chacun des firewalls ce protocole, l’un faisant office à la fois de client sur le FW2 et de serveur sur le FW. A ceci s’ajoute, une clé de chiffrement AES 256 bits de part et d’autre. Cette installation s’est soldée par un test de connectivité entre les deux sites. Ce qui a permis d’intégrer une configuration OpenVPN.

La procédure de configuration étant identique pour les deux PFSENSE, il est décrit ici une procédure.

Une fois connecté à l’interface web de pfsense avec l’utilisateur et mot de passe par défaut, se rendre dans l’onglet VPN>IPSEC puis sur ADD pour ajouter un nouveau tunnel IPSEC. Une fois les paramètres IP renseignés au niveau de l’interface WAN, il suivra une phase d’authentification qui permettra son paramétrage.

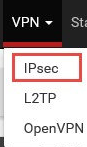


Fig.6 IPSec

* Négociation mode : Agressive
* My identifier : Peer IP address + IP Wan FW1
* Pre-Shared Key : Clé d’authentification qui sera a entrée sur nos 2 FW.
* Encryption Algorithm : AES
* Hash : SHA256

Cette phase une fois terminée, nous avons poursuivi avec la phase deux en renseignant les champs suivants.

* Mode : Tunnel IPv4
* Local Network : LAN + Addresse IP Lan
* Remote network : Network
* Protocol : ESP
* Encryption Algorithms : AES
* Hash Algorithms : SHA256

Ensuite, nous avons permis la connexion au firewall, en activant au niveau des deux pfsenses les protocoles TPC/UDP et dans l’interface **/ WAN** décocher les deux cases **Reserved Network** afin d’autoriser les connexions sur le WAN. Nous nous sommes rendus ensuite sur l’onglet status**>IPSEC** pour mettre en marche le tunnel IPSEC.

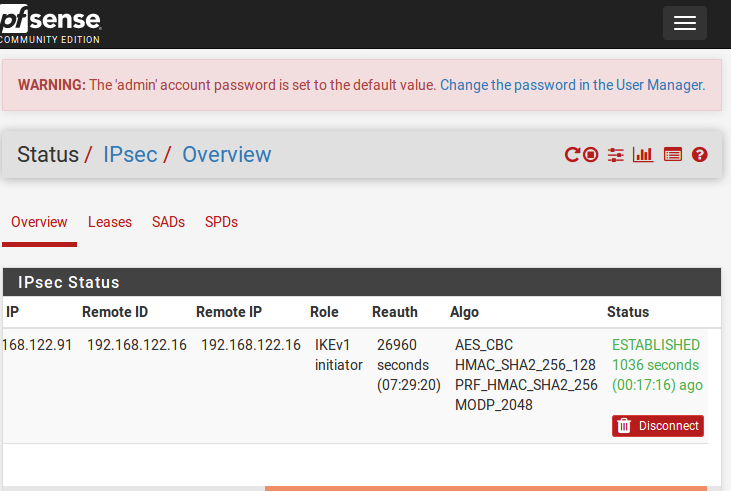


fig.7 ajout IPSec

##### Protocol OpenVPN

Tout comme IPSec, il établit une connexion distante entre sites à la différence qu’ici c’est le client qui contacte en premier le firewall à travers une clé partagée que seul le serveur reconnaitra.

###### Serveur

La solution OPENVPN mise en place ici est une connexion client-serveur. Ainsi, toujours sur IPSEC, se rendre dans VPN>OPENVPN>WIZARD enfin de mettre en place le serveur.

Dans les options de configuration, renseignons les utilisateurs qui seront géré en locale. Ensuite, crée une autorité de certification avec une longueur, une durée de vie, un code ISO du pays, une province, une ville et enfin un nom de société. Ceci fait, il fallut mettre en place un certificat serveur. Viendra, par la suite, le paramétrage de OPENVPN. Nous avons choisi l’interface sur laquelle le serveur va écouter (l’interface WAN), le protocole utilisé, le port d’écoute et enfin une description générique.

Nous continuons notre configuration avec le choix de l’algorithme de hashage dans le cas présent SHA256. Viendra ensuite, la partie tunnel dans laquelle nous avons choisi une adresse pour le réseau virtuel par lequel, les clients passeront pour atteindre notre réseau local distant, et le nombre d’utilisateur maximum autorisé à utiliser le tunnel VPN simultanément. Nous terminerons cette partie de notre configuration en nous rendant, dans l’onglet firewall pour y autoriser le trafic en provenance des clients vers le serveur OPENVPN.

###### Client

Lors de la consignation du serveur OPENVPN, nous avons choisi un mode d’authentification **user mot de passe**. Cependant, il fallut mettre en place les utilisateurs qui pourront se connecter au serveur OPENVPN.

L’installation du paquet OpenVPN-client-export nous a permis d’exporter très facilement notre configuration OPENVPN. Pour le faire, nous nous sommes rendus dans l’onglet **système> packet manager>available packages ,** puis installer le package spécifique.

Une fois l’installation terminer, il fallut se rendre dans système **user manager>group** et rajouté un groupe pour nos utilisateurs nomades. Cette étape terminée, nous avons, exporter notre configuration OPENVPN. Il a donc fallu par la suite installer sur le client un logiciel OpenVPN client et lui attacher le fichier de configuration exporté enfin qu’il puisse s’authentifier et se faire reconnaitre par le serveur.

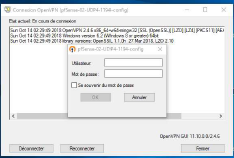


Fig.8 connexion d’un client

### Développement du système de contrôle

Le monitoring ou monitorage est une activité de [surveillance](https://fr.wikipedia.org/wiki/Surveillance) et de [mesure](https://fr.wikipedia.org/wiki/Mesure_physique) d'une activité [informatique](https://fr.wikipedia.org/wiki/Informatique). On parle aussi de [supervision](https://fr.wikipedia.org/wiki/Supervision).

Les raisons de cette pratique peuvent être variées notamment pour des raisons de :

* [Performance](https://fr.wikipedia.org/wiki/Performance), en termes de temps de réponse par exemple ;
* [Disponibilité](https://fr.wikipedia.org/wiki/Disponibilit%C3%A9), indépendamment des performances ;
* D'[intégrité](https://fr.wikipedia.org/wiki/Int%C3%A9grit%C3%A9), l'état des processus sur une machine linux par exemple.

Dans le contexte actuel , travaillant dans un environnement de test, nous nous sommes servis d’un système d’exploitation Ubuntu 16.4 enfin de déployer notre solution.

Lors de notre première élaboration de solution, nous envisagions de mettre dans l’un des réseaux locaux un serveur qui devait faire du monitoring et un autre en externe enfin de faire du failover.

Mais après réévaluation des possibles risques et de perte d’information ou d’indisponibilité de ce serveur ,nous avons recouru à une autre solution qui fut de mettre en place deux serveurs qui nous permettraient de faire du basculement entre les deux si jamais l’un tombait. Cependant, enfin de pousser un peu plus l’idée , il fut judicieux de mettre entre ces deux serveurs une solution appelée pacemaker que nous développerons dans la section suivante. Cela étant, et à des fins plus amélioratives, nous nous sommes posé la question de savoir ce qui se passerait si jamais l’un des serveurs présents en local était stoppé et qu’adviendra-t-il de ce dernier une fois qu’il sera rétabli. D’où, l’idée d’une base de données partagée entre les deux serveurs.

Dans la suite de ce rapport nous développerons les aspects importants et pertinents en ce qui concerne la mise en place d’un tel système de surveillance.

Il convient d’avoir sur chacune des machines que nous nommerons Zabbix serveur1 (ZS1) et Zabbix serveur2 (ZS2) des os ( operating system) Ubuntu 16.04 identiques. La version Xénial de Ubuntu et la base de données MySQL ont été utilisées sur celle-ci.

Mais avant d’entrer dans le vif de l’installation, il convient de faire une présentation de l’utilité de chaque logiciel installé dans les manipulations qui couvriront la supervision d’un telle infrastructure réseau.

#### Zabbix

Zabbix est une plateforme gratuite de supervision et d’alerte en temps réel. Son  
principal objectif est la supervision de l’ensemble d’équipements des réseaux. Il permet  
à l’administrateur de surveiller ses équipements réseaux . Ce logiciel intègre la gestion des pannes et la gestion des performances.

L’architecture Zabbix repose sur quatre composants donc un Zabbix serveur, Zabbix agent, Zabbix frontend, Zabbix proxy et MySQL.

##### Zabbix serveur

Il tourne exclusivement sur linux, et il est chargé de collecter les données, les analyser, réaliser des graphes, les stocker dans une base de données.

##### Zabbix-agent

C’est un logiciel installé sur les équipements qui devront être surveillés. Il est possible de l’installer sur des systèmes d’exploitation tels que Windows, linux, mac etc. Il interagit, collecte, envoie les données récupérées et écoute sur le port 10050 par défaut.

Notons que la collecte d’information se fera auprès des différents périphériques c’est l’exemple pour un switch ou d’un routeur où nous utiliserons le protocole SNMP pour les monitorer

##### Zabbix-front-end

Ce logiciel va permettre à travers une interface web de mettre en forme les données  
(graphiques), consulter les dernières mesures, aussi d'administrer

##### Zabbix proxy

Enfin de décharger les serveurs zabbix(ZS1 OU ZS2), de surveiller les emplacements distants mais surtout de simplifier la maintenance du réseau distribué , nous avons mis en place sur le réseau distant un serveur zabbix proxy enfin de collecter des données de performances et de disponibilité pour le compte des serveurs zabbix .

Nous avons pu remarquer ainsi que l’utilisation d’un proxy est un moyen plus simple d’implémenter une surveillance centralisée et distribuée. Car toutes les données récoltées sont stockées localement avant d’être transmises aux serveurs principaux.

Ainsi, aucune donnée n'est perdue en raison de problèmes de communication temporaires avec le serveur. Les paramètres ProxyLocalBuffer et ProxyOfflineBuffer dans le [fichier de configuration](https://www.zabbix.com/documentation/3.0/manual/appendix/config/zabbix_proxy) du [proxy](https://www.zabbix.com/documentation/3.0/manual/appendix/config/zabbix_proxy) contrôlent la durée de conservation locale des données.

Il est bien de noter que , le proxy Zabbix est un collecteur de données. Il ne calcule pas les déclencheurs, ne traite pas les événements et n'envoie pas d'alertes.

#### Intégration de la base de données partagées

Pour des raisons de disponibilités d’informations, il est ici installé dans un serveur différent des deux autres une base de données donc le moteur utilisé ici est MYSQL server.

Sur le serveur nommé zabbix data base (ZSDB) dédié au partage de la base de données , la collecte et la sauvegarde de l’état de la BD des différents serveurs zabbix1 et zabbix2. Jusqu’ici se faisait en local. Nous avons ainsi procédé à l’installation d’une base de données secondaire ou de backup nommée mariadb.

apt Install mariadb-server.

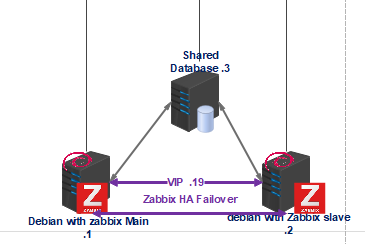
Nous l’avons sécurisée en créant un nouvel utilisateur et un mot de passe pour se connecter à la base de données. Il fallut ensuite se rendre sur l’un des serveurs zabbix de surveillance enfin de transférer le script du fichier de base de données local vers la base de données partagée.

Bien évidemment, une connexion cliente ssh est nécessaire pour ce type de transfert car certaines informations d’authentification seront demandées pour se connecter au serveur distant.

#### Haute disponibilité

En effet, le failover dans notre cas est un dispositif mis en place entre nos deux serveurs formant des nœuds où chacun fonctionne indépendamment de l’autre.il aura pour objectif, d’assurer la haute disponibilité et fonctionne selon ce principe ; quand un nœud tombe en panne ou est indisponible pour une quelconque raison, un autre nœud du cluster prend la relève et assure le relais pour maintenir la disponibilité du serveur en entendant le rétablissement de l’autre nœud.

Pour le mettre en pratique dans notre situation, il fallut procéder à l’installation de pacemaker et de corosysnc.

 Fig.10 failover

#### Pacemaker

C’est un gestionnaire de cluster haute disponibilité. Il est chargé de démarrer, arrêter et de superviser les ressources d’un cluster. Il a pour rôle de basculer entre deux ou plusieurs machines à travers une IP de basculement préalablement configuré. En outre, c’est cette IP de basculement qui est renseigné dans la configuration des deux Zabbix serveurs permettant ainsi la liaison pacemaker Zabbix.

Pour qu’une telle installation fonctionne, nous avons référé ces champs dans zabbix enfin de pouvoir identifier et de renseigner chaque basculement.

* Zabbix1 : premier système de basculement ( zabbix1.example.com ).
* Zabbix2 : deuxième système de basculement ( zabbix2.example.com) .
* Zabbix - IP de basculement qui est une adresse IP flottante du réseau sur lequel, sont situés mes deux serveurs.

#### Corosync

L’installation de pacemaker s’appuyant sur d’autres logiciels tels heartbeat ou corosync pour agir et surveiller des postes, il nous fallut permettre aux deux machines de s’envoyer des informations d’où l’usage de corosync

# Systèmes de monitoring

Pour effectuer une surveillance sur le réseau , l’agent zabbix devait être présent et activé sur chacun des périphériques du réseau. De même, il fallait aussi installer et activer en parallèle les services à monitorer. Pour par la suite l’intégrer au système de surveillance zabbix server.

Le schéma ci-dessous permet d’observer au préalable le flux d’information dans Zabbix.

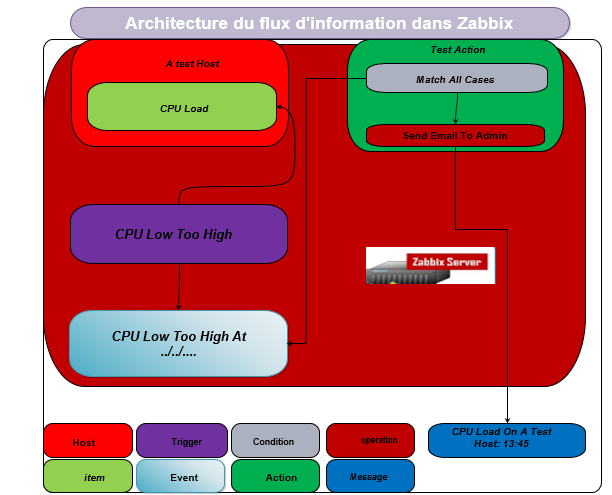


Fig.11

#### Description

Dans cette partie, il est détaillé le flux d’information dans Zabbix. Pour parvenir, on passe par les étapes tel que l’intégration d’un hôte au serveur, et l’acheminement des informations.

##### Intégration d’un hôte à monitorer

Les hôtes Zabbix sont des périphériques que nous souhaitons surveiller (serveurs, commutateurs, poste de travail etc…). La création d’un hôte est l’une des premières tâches de surveillance sur notre serveur Zabbix. À noter que, les hôtes peuvent être organisés dans Zabbix par groupe. L’ajout d’un hôte nécessite une bonne documentation de l’environnement Zabbix enfin de mieux se repérer au moment d’élaborer sa séquence de surveillance. Il est important de bien assigner l’hôte au bon groupe tout en renseignant son nom et son adresse IP. Cela permettra la liaison entre l’agent et le serveur. Mais avant, les informations du serveur devront être préalablement renseigné sur l’agent. Cette figure présente un aperçu des hôtes intégré dans Zabbix serveur.

Cependant, zabbix dispose de Template ou modèles donc l’utilisation reste un excellent moyen de faciliter sa maintenance, de même qu’un ensemble d’entités (éléments, déclencheurs, graphiques, applications, écrans, règles de découverte et scénarios Web) pouvant être affecté à un modèle, ce qui permet ainsi une gestion efficace de milliers de périphériques.

##### Intégration d’un système d’alerte

Le système d’alerte est un ensemble de mesure mis en place pour assurer la disponible d’un service ou d’un périphérique surveillé.

A ce niveau de notre projet, divers services et ressources ont été configuré et sont présent dans l'interface Zabbix. Il fallut dont mettre en place un système d’alerte qui s’organiserait en processus logique pour :

* Tester des services et collecte de données (élément=I**tem**) ;
* Générer un évènement en réaction à une certaine valeur ou donnée remontée par un item;
* Envoyer des alertes (notifications), en fonction d’évènements précis générés par des triggers.

Mais comment cela fonctionne-t-il ensemble ?

Dans notre installation de serveur Zabbix, nous avons créé un hôte (A Test Host), qui contient un élément (CPU Load) ,un déclencheur référençant cet élément et une action. Chaque fois que l'expression du déclencheur correspond à la valeur de l'élément en cours, il passe à l'état **PROBLEME**. Lorsqu'il ne correspond plus, il revient à un état **OK.** Chaque fois que le déclencheur change d'état, un événement est généré.

# Difficultés rencontrées

La réalisation de ce projet à surgir plusieurs problèmes parmi lesquelles :

* L’établissement de la haute disponibilité au niveau du pacemaker ;
* La mise place de la base de données partagées mais plus précisément au niveau de la connectivité IP ;
* Le transfert des clés publics SSH entre machine car système réaliser en environnement virtuel.

# Améliorations possibles

Bien que le système soit mis en place pour répondre au cahier de charge, il reste bon nombre de fonctionnalité à incorporer dans celui-ci. Il s’agit par exemple de :

* La mise en place d’un système rapport fournissant un état sur les statistiques du réseau tous les mois ou semaines ;
* La réflexion sur la mise en place d’un failover avec un nœud sur un serveur externe(cloud) ;
* La mise en place d’un failover sur la base de données partagées ;
* La mise en place d’un failover sur Zabbix proxy.

# Recommandations

Au travers de ce rapport, aucun script n’a encore été mis en place durant. En revanche, pour la procédure d’installation ou de déploiement, est héberger sur le lien ci-dessous.

-------------------------------liens du script ici --------------------------------------------------------------------------------

Il est impératif de les exécuter dans l’ordre décrit .

# Conclusion

Le sujet portait sur l’Implémentation **d’un système de surveillance et d’alerte externe d’une infrastructure réseau modulaire**. Ceci permettant de réduire le temps d’intervention lors de certaines défaillances systèmes. A travers les outils tel que Zabbix, GNS3, VMWARE, corosync, pacemaker, ce thème nous a permis de mettre en place un système d’information (SI) dont les différentes phases sont :

* Études comparatives des outils de surveillances ;
* Mise en place des technologies ;
* Développement du système de contrôle et du système de monitoring.

Dans l’optique de notifier les personnes chargées de l’administration du réseau, un système de mailing a été mis en place permettant de nous alerter lorsque survient une défaillance système.

En définitif, bien que ceci réponde à la ligne directive fixée en introduction, il reste encore à ce jour un grand travail à effectuer au niveau du système d’alerte qui est celui de la mise en place d’un système de backup (sauvegarde automatique) de notre système de monitoring vers le serveur NAS prévus pour la contenance de ces fichiers.

# Bibliographie

https://www.supinfo.com/articles/single/2082-mise-place-cluster-basculement-failover-windows-server-2012-r2 (failover)

https://wiki.monitoring-fr.org/zabbix/zabbix-snmp-host (snmp)

https://eric.bugnet.fr/superviser-mon-serveur-avec-zabbix/ (zabbix et item)

http://shyrrow.free.fr/wp-content/uploads/Mise-en-place-zabbix.pdf (zabbix carte)

https://www.supinfo.com/articles/single/3318-mise-place-cluster-avec-heartbeat-debian (failover avec haerdbirt)

https://www.sebastien-han.fr/blog/2011/07/04/introduction-au-cluster-sous-linux/ (failover avec pacemacker)

https://www.tecmint.com/setup-high-availability-clustering-in-centos-ubuntu/ (avec nginx)

https://www.tecmint.com/what-is-clustering-and-advantages-disadvantages-of-clustering-in-linux/

https://www.howtoforge.com/tutorial/how-to-configure-high-availability-and-network-bonding-on-linux/ (failover linux plus test)

(zabbix fail over)

https://zabbixonly.com/how-to-configure-high-availability-ha-zabbix-server-on-ubuntu-18-04/

http://yallalabs.com/linux/how-to-configure-a-high-availability-zabbi-server-using-pacemaker-on-centos-7-rhel-7/

https://bugs.launchpad.net/ubuntu/+source/pcs/+bug/1640923 (correction des bugs de pacemaker)

https://docs.microsoft.com/en-us/sql/linux/sql-server-linux-configure-multiple-subnet?view=sql-server-2017 (différent subnet)

(Agent windows)

http://yallalabs.com/windows/how-to-install-zabbix-agent-3-4-and-add-windows-host-to-zabbix-monitoring/

(Switch and router snmp monitor)

https://securitywing.com/configure-snmp-zabbix-server-debian-monitor-cisco-switches/

(Freenas monitor)

https://docs.j7k6.org/freenas-zabbix-agent/

https://share.zabbix.com/storage-devices/freenas/freenas-11 (import Template snmp)

https://techexpert.tips/category/zabbix/ (toute la surveillance zabbix)

(Zabbix and Gmail)

https://www.tecmint.com/configure-zabbix-to-send-email-alerts-to-gmail/

(Zabbix and sms)

https://techexpert.tips/zabbix/zabbix-sms-notification-setup/

https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-zabbix-to-securely-monitor-remote-servers-on-centos-7 (connexion agent zabbix Secure)

https://linoxide.com/monitoring-2/install-zabbix-monitoring-server-agent-centos-7/ (Install zabbix)

https://linoxide.com/monitoring-2/configure-zabbix-proxy-external-monitoring-centos-7/ (Install proxy CentOS)

http://yallalabs.com/linux/how-to-install-and-configure-zabbix-proxy-3-4-on-ubuntu-16-04-lts/ (Install proxy Ubuntu)

https://books.google.be/books?id=eyyFDwAAQBAJ&pg=PA588&lpg=PA588&dq=zabbix+server+proxy+same+machine&source=bl&ots=XFuEY815EE&sig=ACfU3U2X4mc7v2\_jRMi2OcmI7MoEZUP9OA&hl=fr&sa=X&ved=2ahUKEwj3jJXnsdXiAhUEb1AKHanUBoAQ6AEwCXoECAkQAQ#v=onepage&q&f=true (livre)

https://www.youtube.com/watch?v=zfv-UIxTbew (sauvegarde dans gns3)

https://jackal777.wordpress.com/2011/03/05/ubuntu-10-10-bind-round-robin-howto/ (Dns round robin)

(Backup PfSense)

<https://blogmotion.fr/systeme/script-backup-pfsense-configuration-16496>