Ministère de l'Enseignement Supérieur,
de la Recherche Scientifique et des Technologies
de l'Information et de la Communication
Université Virtuelle de Tunis

UVT
جامعة تونس الافتراضية
UNIVERSITÉ VIRTUELLE DE TUNIS

RAPPORT
DE STAGE DE FIN D'ETUDES

Pour l'obtention de la
«Licence Appliquée en Sciences et Technologies de l'Information et de Communication (LASTIC)»

Présenté par:

Ellouzi Dhekra

Mise en place d'un centre d'opérations réseau NOC à Topnet

Soutenu le : 03/07/2017

Devant le jury :

Président : Mr Hassen Seddik

Encadreur : Mme Rim Cherif, maître de conférences à l'iset'Com

Rapporteur : Mr Ezzeddine Ben Braiek

Année Universitaire : 2016 / 2017

www.uvt.rnu.tn
14, Rue Yahia Ibn Omar - 1082 Mutuelleville - Tunis
Tél. : +216 71 28 99 81 / +216 71 89 17 31 - Fax : +216 71 89 26 25

Dédicaces

A ma mère et mon père mes professeurs de toujours,

A mon très cher mari, mes enfants

pour leurs amours et sacrifices.

A mes proches amis et toute ma grande famille,

pour leurs soutiens et encouragements.

A toutes les personnes qui me connaissent de près ou de loin.

Remerciements

Je tiens à adresser mes sincères remerciements à toutes les personnes qui m'ont aidée dans la réalisation de ce travail de fin d'études.

Je remercie particulièrement :

Mme Rim Cherif, maître de conférences à l'Iset'Com, pour son encadrement, le temps qu'elle a consacré tout au long de ce projet, sa patience, sa disponibilité et ses conseils qui nous ont aidés dans l'accomplissement de ce travail.

Mr Saifeddine Ameur, Responsable Département Services Internet à Topnet, pour les conseils utiles qu'il n'a cessé de me prodiguer, les orientations et l'intérêt qu'il m'a montré durant la progression de ce travail.

Je voudrais aussi adresser mes plus vifs remerciements à toute l'équipe pédagogique de l'Université Virtuelle de Tunis et les intervenants professionnels responsables de la formation LASTIC.

Finalement, j'adresse mes sincères remerciements à tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce projet, qu'ils trouvent ici l'expression de ma profonde reconnaissance, ainsi qu'aux membres du jury pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail.

Table des matières

Table des figures

Introduction Générale

De nos jours, on reconnait une forte croissance de l'intégration et de l'utilisation des réseaux de communication et des systèmes d'information au sein de l'activité des entreprises. Toutefois, avoir un système informatique performant, est devenu une priorité.
En cas de panne survenue dans leur système d'information, les entreprises peuvent perdre une grande quantité de leur profit ce qui peut causer dans certains cas leur faillite. Ainsi, chaque organisme doit assurer la disponibilité de ses systèmes informatiques pour pouvoir garantir la continuité de leurs services et le bon déroulement de leurs activités.

Certes, tous les fournisseurs de services informatiques (FSI) de nos jours, prennent en compte l'aspect haute disponibilité dans la conception et l'implémentation de leurs solutions et tentent de concevoir des produits tolérants aux pannes, qui fonctionnent sans interruption et à plein temps. Cependant, ces efforts ne garantissent pas toujours le bon fonctionnement des systèmes. Les pannes sont cependant, toujours présentes.

Ainsi, la détection des pannes à l'avance est devenue une nécessité pour les administrateurs des systèmes informatiques, pour pouvoir réagir à temps et minimiser les pertes, chose qui est de plus en plus difficile, surtout dans des systèmes complexes contenant des centaines d'équipements.

L'approche la plus fiable serait de pouvoir détecter et analyser les pannes en avance, avant même que le client ne le réclame et sans qu'il ne s'aperçoive d'une dégradation du service.

Actuellement, l'enjeu pour les FSI est de pouvoir garantir la continuité de leurs services, tout en gardant les pannes/incidents transparents pour leurs clients. D'où, le besoin d'un centre d'opération réseau en anglais, Network Operations Center, abrégé NOC, qui assure la surveillance des pannes, l'analyse des données et la gestion des incidents ou dysfonctionnement.

Un centre NOC doit assurer les fonctionnalités suivantes :

- Surveiller la plateforme des services internet et réseau 7j/7 et 24/24.
- Assurer un système de ticketing et notifications.
- Gérer les incidents et les alertes : diagnostic et suivi jusqu'à la résolution des problèmes.

Afin d'avoir une vue globale sur son infrastructure système et réseau, d'anticiper et de réagir à temps aux éventuels incidents qui peuvent toucher sa plateforme de services internet et ses équipements réseau, TOPNET a décidé de mettre en place son propre centre NOC. Ce qui fera l'objet de ce projet de fin d'étude.

Ce rapport est structuré en trois chapitres. Le premier chapitre de ce rapport présente l'état actuel du service monitoring, une brève description des différents composants du centre d'opération réseau ainsi que son intérêt et enfin ses spécifications et besoins.
Ensuite, le deuxième chapitre présente l'architecture générale du projet et expose ses composants.
Le troisième chapitre expose les étapes d'installation et une présentation des différentes interfaces des applications déployées ainsi de quelques cas pratiques d'un centre NOC. Je terminerai ce rapport par une conclusion générale, synthétisant les résultats et les perspectives de ce travail.

Chapitre 1
Présentation du cadre du projet

1.1 Introduction

Dans ce chapitre, nous allons présenter quelques notions de base pour la mise en place d'un centre d'opérations réseau NOC (en anglais, Network operations center, abrégé NOC). C'est un service de surveillance des évènements, analyse des données, gestion des incidents ou dysfonctionnement [1].
Puis, on mettra en évidence l'intérêt de ce projet tout en déduisant son utilité pour Topnet.
Et finalement, nous allons présenter les objectifs de notre projet, ce qui nous amène à identifier les possibilités du système NOC et les besoins que nous exposerons sous forme de diagrammes de cas d'utilisations globales et détaillés.

1.2 Principe de fonctionnement d'un centre NOC

Aujourd'hui, les organisations s'efforcent constamment d'adopter de nouvelles technologies, en mettant à profit l'infrastructure de pointe pour les fonctions informatiques critiques de l'entreprise telles que le RH, le marketing et les ventes.
Dans la concurrence actuelle du marché, même un temps d'indisponibilité réduit (causé par des pannes d'infrastructure) a un énorme impact sur les revenus et la réputation de l'entreprise. Les fournisseurs de services informatiques ont reconnu ce défi et ont commencé à offrir un service plus fiable, évolutif et hautement disponible. Cela est assuré par le soutien de l'infrastructure sous l'acronyme «NOC», centre d'opérations du réseau.

1.2.1 Qu'est-ce qu'un centre NOC

Un centre d'opérations du réseau (en anglais, Network operations center, abrégé NOC) est un service de surveillance des évènements, analyse des données, gestion des incidents ou dysfonctionnement [1]. La figure 1 représente un vidéo-wall et quelques outils de supervision et de remontée utilisés dans un centre NOC.

Figure 1. Exemple d'un centre NOC [2].

Le centre NOC est chargé principalement de :

- Surveiller l'infrastructure des services internet et réseau 7j/7 et 24/24.
- Recevoir, reconnaître et traiter les événements.
- Assurer un système de ticketing et notifications par e-mail et SMS.
- Gérer les incidents et les alertes : diagnostic et suivi jusqu'à la résolution des problèmes.
- Améliorer la disponibilité des services : réduire les interruptions d'activités en raison des disfonctionnements ou incidents.
- Offrir des services, d'implémentation et de support pour les services de la direction technique.

Il peut également aider à planifier, déployer et exploiter l'infrastructure de base [4].

1.2.2 Principaux composants

Afin de mettre en œuvre un service efficace et fiable, un fournisseur des services internet est tenu de gérer son infrastructure avec une véritable discipline en utilisant une structure cohérente pour la gestion des informations recueillies. Cette structure englobe quatre éléments essentiels (comme illustré dans la figure 2) : un système de supervision, un système de visualisation, un système de ticketing et un système ou guide de gestion d'incidents.

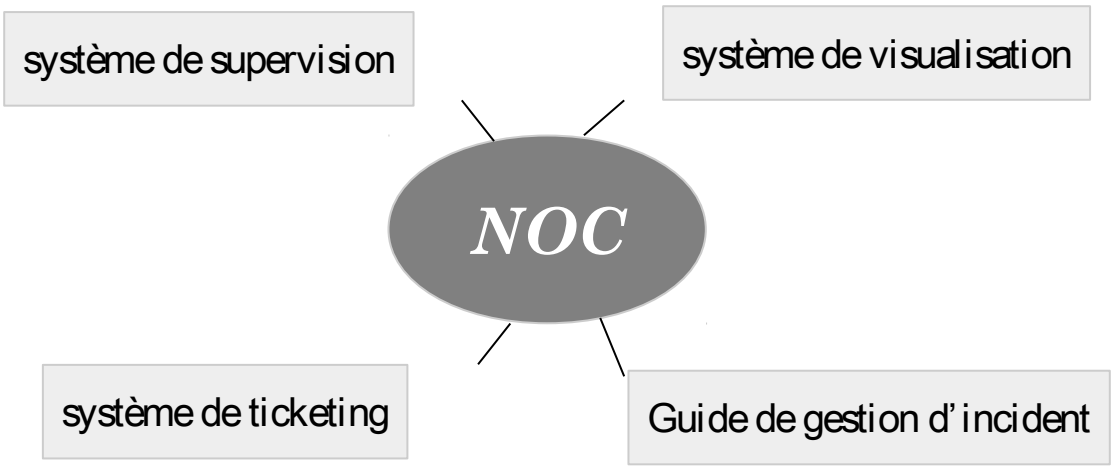


Figure 2. Composants d'un centre NOC

1.2.2.1 Système Monitoring (outil de supervision)

La supervision de réseaux peut être définie comme l'utilisation de ressources réseaux adaptées dans le but d'obtenir des informations (en temps réel ou non) sur l'utilisation ou les conditions des réseaux et de leurs éléments afin d'assurer un niveau de service garanti, une bonne qualité et une répartition optimale.

Le système de supervision assure la :

-Visualisation des graphes des éléments à superviser.

-Possibilité d'avoir des rapports à générer,

-Réception des alertes par Email, SMS,…

-Diagnostic (erreur récurrentes, corrélation)

-Planification de capacité grâce à deux types de mesure : mesures actives (Ping, trace route, Telnet, snmpget ifStatus) et mesures passives (traps SNMP, logs syslogs,...).

Exemple d'outils de supervision : PRTG (Paessler Router Traffic Grapher), CACTI (Cache Access and Cycle Time Model), MRTG (Multi Router Traffic Grapher),….

1.2.2.2 Système de visualisation

Les outils de supervision fournissent souvent une interface basique de visualisation des données. Néanmoins, Ils peuvent être suffisants comme étant un moyen de supervision classique, mais dès que l'on veut construire des tableaux de bord partagés et analytiques, il est suggéré de déployer une solution spécialisée pour l'affichage d'un mur de tableaux de bord. Cela est surtout nécessaire pour le centre NOC, qui impose une solution de dashboarding.

Exemple d'outils de visualisation : Grafana, kibana, Graphite, Data Studio,…

1.2.2.3 Système tickets et reporting

Un système tickets ou système de suivi de problèmes (de l'anglais issue tracking system) est un logiciel qui permet d'aider les utilisateurs et les développeurs à améliorer la qualité des services.
Les utilisateurs soumettent leurs tickets à l'aide de l'outil. Les administrateurs système sont alors engagés à traiter ces tickets afin de résoudre les problèmes signalés.
Ce type d'outils était conçu pour suivre les incidents ou anomalies d'un projet. Désormais, certains logiciels de suivi de problèmes sont configurables et permettent de gérer tous types de tickets ou alertes : tâches, demandes de support, exigences,...
Le système tickets permet de :
- Créer un ticket pour chaque problème signalé.
- Assigner le problème au 2ème niveau ou met à jour le statut des tickets.
- Favoriser l'archivage de tout dysfonctionnement sur du long terme.
- Donner des analyses statistiques (incidents/période, type, temps, moyen de résolution, etc...)
- Servir de base de connaissances (knowledge base)
Exemple d'outils de Ticketing : RT (Request Tracker), OTRS (Open-source Ticket Request System) , GLPI (gestionnaire libre de parc informatique), …

L'équipe du centre d'opération se charge de :
- Détecter les anomalies et analyser les graphes et données (24x7)
- Ouvrir des tickets d'incidents pour suivre les problèmes
- Procéder au diagnostic préliminaire (1 st level)
- Assigner le problème à l'équipe, et met à jour le statut des tickets.
- Assure le suivi du problème jusqu'à sa résolution.

1.2.2.4 Système de gestion d'incident

Gestion des incidents : « l'objectif de la gestion des incidents est : Restaurer aussi vite que possible le fonctionnement normal des services et minimiser l'impact négatif sur les activités métiers et s'assurer ainsi que les meilleurs niveaux de qualité de service et de disponibilité sont maintenus. » [5]
Un incident est : « Tout événement qui ne fait pas partie du fonctionnement standard d'un service et qui cause, ou peut causer, une interruption ou une diminution de la qualité de ce service.» [5]
Le document de gestion d'incident permet de constituer un guide de mise en place d'un système de gestion d'incidents assurée par le service NOC.

Ce document définit les indicateurs clés de performance (KPI) ainsi que les éléments essentiels de la gestion des remontées : La détection, le diagnostic, la catégorisation la qualification, la procédure de remontée et la résolution pour chaque type d'incident. Il nécessite des mises à jour régulières pour garantir une base de connaissance cohérente.

1.3 Etude de l'existant

1.3.1 Monitoring actuel

Actuellement, la surveillance des services et des équipements de la plateforme services internet et réseau est assurée par l'outil de monitoring Zabbix (service mail, hosting, service AAA, DNS, monitoring, trafic réseau,...) et les équipements (machines physiques, serveurs virtuelles, switchs, onduleurs, pare-feu, équipements de stockage, serveurs de gestion des logs,...).

La figure 3 présente le tableau de bord de notre outil de supervision actuel.

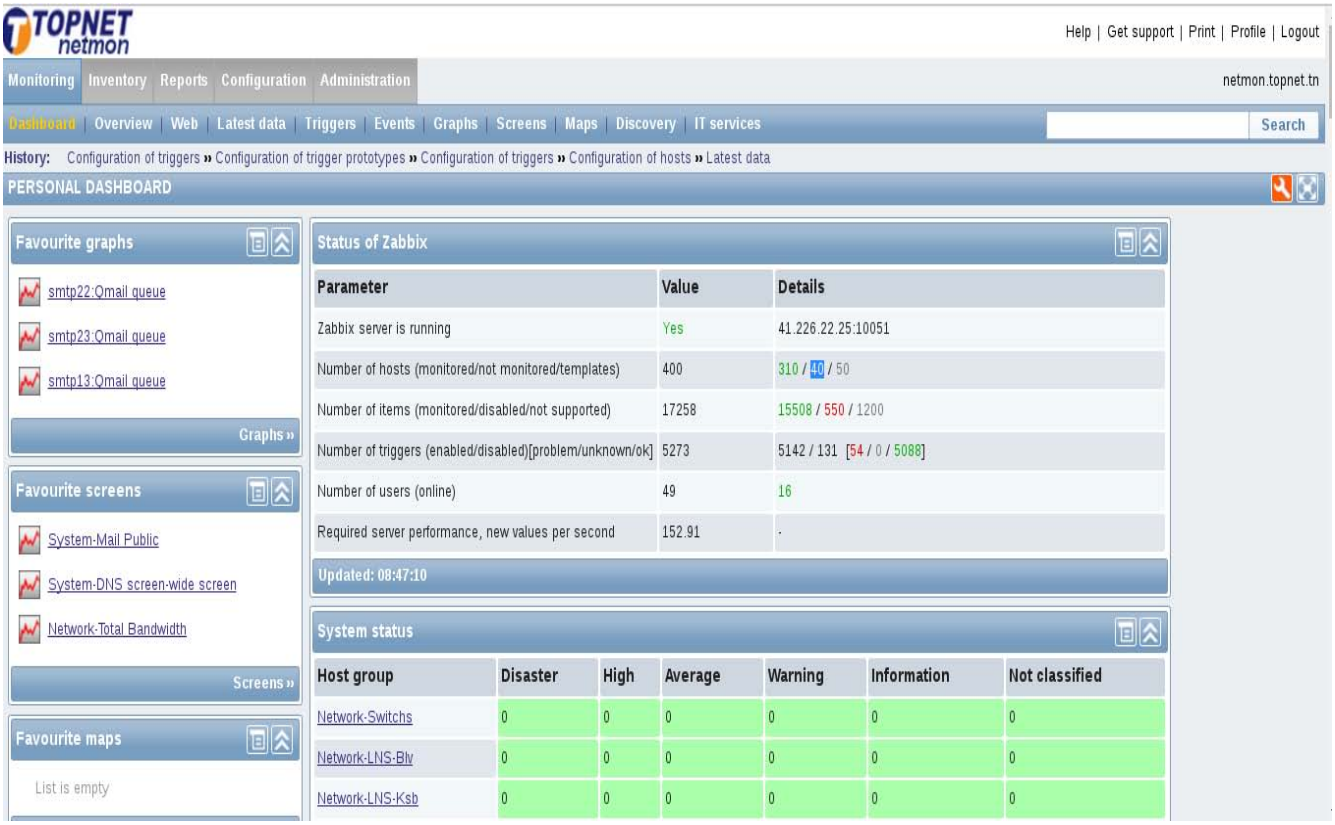


Figure 3. Tableau de bord de l'outil de monitoring.

Ainsi, chaque service surveille les équipements et les services concernés depuis des écrans conçus sur cet outil. La figure 4 représente l'écran conçu pour surveiller l'état de la queue des emails au niveau des serveurs mails.

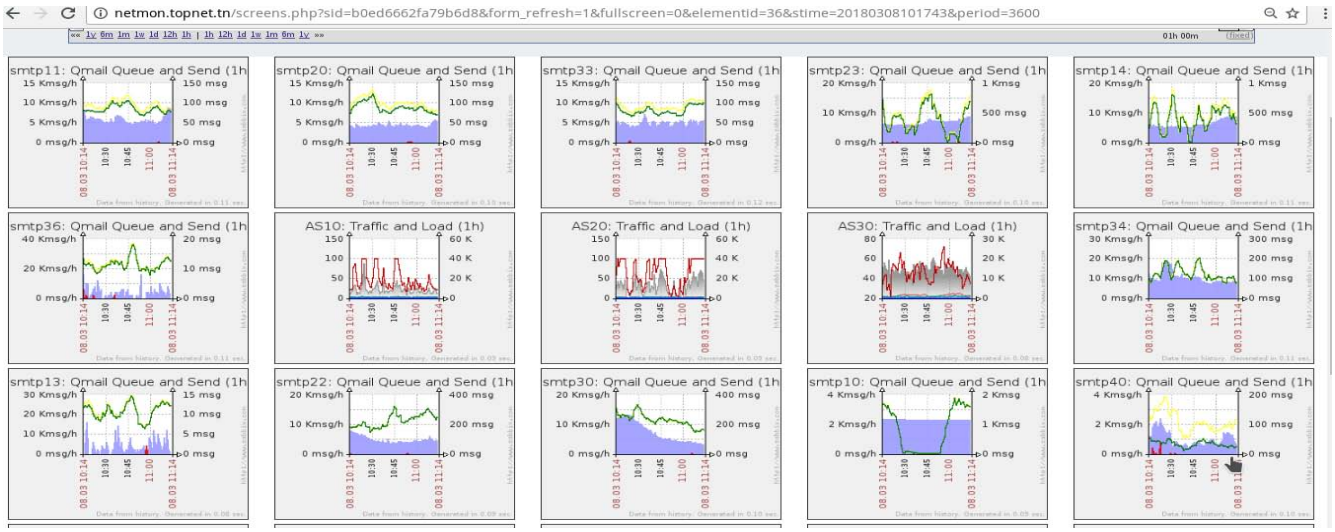


Figure 4. État de la queue des emails au niveau des serveurs SMTP.

Par la suite, chaque équipe doit vérifier l'état des services, les performances des équipements et des serveurs en se référant aux informations collectés par cet outil. Des déclencheurs d'alertes sont aussi configurés afin de signaler tout disfonctionnement ou incidents survenus à cette équipe.

Ainsi, la remontée est effectuée par mail ou SMS et non pas par un outil de ticketing. Ceci implique :

-Des fautes lors de la déclaration et qualification des incidents.

-Des remontées sans analyse et sans 1er diagnostic.

-Les procédures de la remontée ne sont pas mises à jour.

-La réception des fausses alertes.

-Des difficultés pour chaque équipe à détecter des corrélations et des erreurs répétitifs.

-Manque de reporting sur l'état des performances système.

-Pas de vue globale sur l'état de la plateforme.

1.3.2 Solution NOC et SOC

Il y a plusieurs façons par lesquelles une entreprise lutte contre un dysfonctionnement ou dégradation du service. Cela se fait généralement dans un Centre des opérations du réseau (NOC) ou un Centre des opérations de sécurité (SOC).

Le terme SOC désigne ainsi une plateforme dont la fonction est de fournir des services de détection des incidents de sécurité, mais aussi de fournir des services pour y répondre. Le centre de sécurité va donc collecter les événements (sous forme de logs notamment) remontés par les composants de sécurité, les analyser, détecter les anomalies et définir des réactions en cas d'émission d'alerte.

Bien que leurs tâches principales semblent être similaires (l'identification, diagnostic, la gestion d'incident), les types de problèmes et leur impact sont considérablement différents [3].

En effet, le fonctionnement du service NOC consiste à respecter les ententes de niveau de service (SLA) et à gérer les incidents de manière à réduire les temps d'arrêt, c'est-à-dire à mettre l'accent sur la haute disponibilité. Tandis que, l'efficacité d'un centre SOC est mesurée en fonction de leur capacité à protéger la propriété intellectuelle et les données sensibles - un accent sur la sécurité.

En outre, l'exigence en termes de qualité et haute disponibilité de la plateforme des services internet revêtent une importance cruciale pour la mise en place d'un centre NOC [6].

1.3.3 Intérêt NOC

En tant que leader des fournisseurs Internet avec 51.7%, l'objectif primordial de la direction technique de Topnet est d'optimiser la qualité de service et atteindre la plus haute disponibilité de la plateforme et services offerts à ses clients. Cela exige non seulement l'amélioration de la surveillance des disfonctionnements mais aussi la gestion de la qualité améliorée afin de réduire l'impact des incidents sur les activités, obtenir une meilleure utilisation des ressources, et de garantir la satisfaction des clients.

Ainsi, le centre NOC permet à Topnet :

-Avoir un centre NOC qui centralise la supervision (24/24, 7/7) pour toutes les plateformes des différents services de la direction technique de Topnet.

-Avoir une vision globale sur toute la plateforme Topnet, son état, son dimensionnement, et son évolution.

-Centraliser à la fois la supervision et la remontée

-Assurer un reporting complet sur les différents alertes et incidents sur la plateforme.

-Standardiser les procédures de la remontée

-Minimiser le temps de gestion, diagnostic et résolution des incidents.

1.4 Besoins fonctionnels

Les besoins fonctionnels ou besoin métiers représentent les actions que le système NOC doit exécuter. Il ne devient opérationnel que s'il les satisfait.

Ce projet doit couvrir principalement les besoins fonctionnels suivants :

- Visualiser un écran global sur les activités des différents services de la plateforme des services internet et réseau.
- Afficher des écrans dédiés à chaque service sur l'outil de supervision.
- Créer, afficher et déclencher des alertes pour les objets existants sur cet outil.
- Création et suivi de tout dysfonctionnement jusqu'à sa résolution.
- Avoir une mise à jour régulière de la procédure de remontée.
- Avoir un reporting journalier sur l'état de la plateforme.
- Avoir un reporting journalier sur l'activité de la gestion de la remontée.

- Affichage de l'historique complet pour chaque composant de l'architecture de la plateforme.

1.5 Besoins non fonctionnels

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du service mais plutôt identifient des contraintes internes et externes. Les principaux besoins non fonctionnels de notre projet se résument dans les points suivants :

-L'ergonomie : offrir des interfaces conviviales et faciles à utiliser compatibles avec n'importe quel système d'exploitation. Ces interfaces doivent être compréhensibles, et bien organisées.

-La sécurité : respect de la confidentialité des données ; gestion des droits d'accès.

- La qualité de service : minimiser la durée de la résolution de dysfonctionnement ; améliorer les indicateurs de performances KPI.

Besoins optionnels

Plus d'options au niveau de l'outil de supervision :

- cryptage des données envoyées au serveur par PSK (pre-shared key)
- déclenchement des alertes par SMS.
- plus de fonctionnalités et mesures par la migration à une version récente de l'outil de supervision.

1.6 Les diagrammes des cas d'utilisation

Les diagrammes des cas d'utilisation montrent les interactions fonctionnelles entre les acteurs et le système à l'étude.

- **Acteur** : rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.
- **Cas d'utilisation (use case)** : ensemble de séquences d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier.
- **Association** : utilisée dans ce type de diagramme pour relier les acteurs et les cas d'utilisation par une relation qui signifie simplement « participe à ».
- **Inclusion** : le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifié dans ses enchaînements.
- **Extension** : le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l'extension.
- Généralisation : les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des relations spécifiques supplémentaires avec d'autres acteurs ou cas d'utilisation [11].

1.6.1 Identification des acteurs

Un acteur représente un rôle joué par une personne qui interagit avec le système NOC.

Par définition, les acteurs sont à l'extérieur. Les acteurs se recrutent parmi les utilisateurs du système et aussi parmi les responsables de sa configuration et de sa maintenance. D'où, les acteurs potentiels qui risquent d'interagir avec le système NOC sont :

- **L'utilisateur Technicien NOC** : il est responsable de superviser tout l'ensemble de composants de l'architecture plateforme. Cet utilisateur ne consulte que les graphes et les alertes. Cet utilisateur ne modifie pas les objets des composants de l'architecture plateforme. C'est lui qui doit créer, ajouter un commentaire, fermer un ticket pour chaque alerte ou dysfonctionnement apparu. Il assure le reporting essentiel. Il consulte la procédure de remontée.
- **L'utilisateur Responsable NOC** : Celui-ci valide la création et la modification des objets des composants de l'architecture plateforme signalées par les équipes du service plateforme et réseau. Il a une visibilité totale sur tous composants de l'architecture plateforme. Il a pour tâches de gérer les outils de supervision et de ticketing. Il spécifie les utilisateurs et les droits de chacun. Il met à jour la procédure de remontée.
- **L'utilisateur service 2ème niveau** : Il ne consulte que les objets et écrans relatives à son service. Il consulte et ajoute des commentaires aux tickets qui lui sont assignées.
- **L'utilisateur service 3ème niveau** : Il a le droit de consulter tous les objets des composants de l'architecture plateforme. Il consulte et ajoute des commentaires aux tickets qui lui sont assignées.

1.6.2 Les différents cas d'utilisation

L'étude de cas d'utilisation a pour objectif de déterminer ce que chaque utilisateur attend du système. La détermination du besoin est basée sur la représentation de l'interaction entre l'acteur et le système.

1.6.2.1 Cas d'utilisation d'un centre NOC

- **Supervision et consultation** : superviser l'ensemble de composants de l'architecture plateforme et consulter les objets des composants de l'architecture plateforme selon le droit d'accès autorisé.
- **Gestion de l'outil de ticketing** : permet de créer, mettre à jour ou fermer les tickets pour chaque alerte ou dysfonctionnement apparu.
- **Mise à jour des tickets** : affiche et ajoute un commentaire à un ticket ouvert.
- **Reporting** : permet d'afficher et imprimer les tableaux de bord, les statistiques et l'historique de chaque ticket.
- **Consulter la procédure de remontée.**
- **Gestion de l'outil de supervision** : permet à l'administrateur d'ajouter, modifier ou supprimer ou consulter l'outil de supervision.
- **Gestion d'incident** : permet de mettre à jour la procédure de remontée.
- **Gérer les utilisateurs et les droits d'accès** : permet à l'administrateur d'ajouter ou supprimer ou modifier ou consulter un utilisateur pour les outils de supervision et de ticketing.

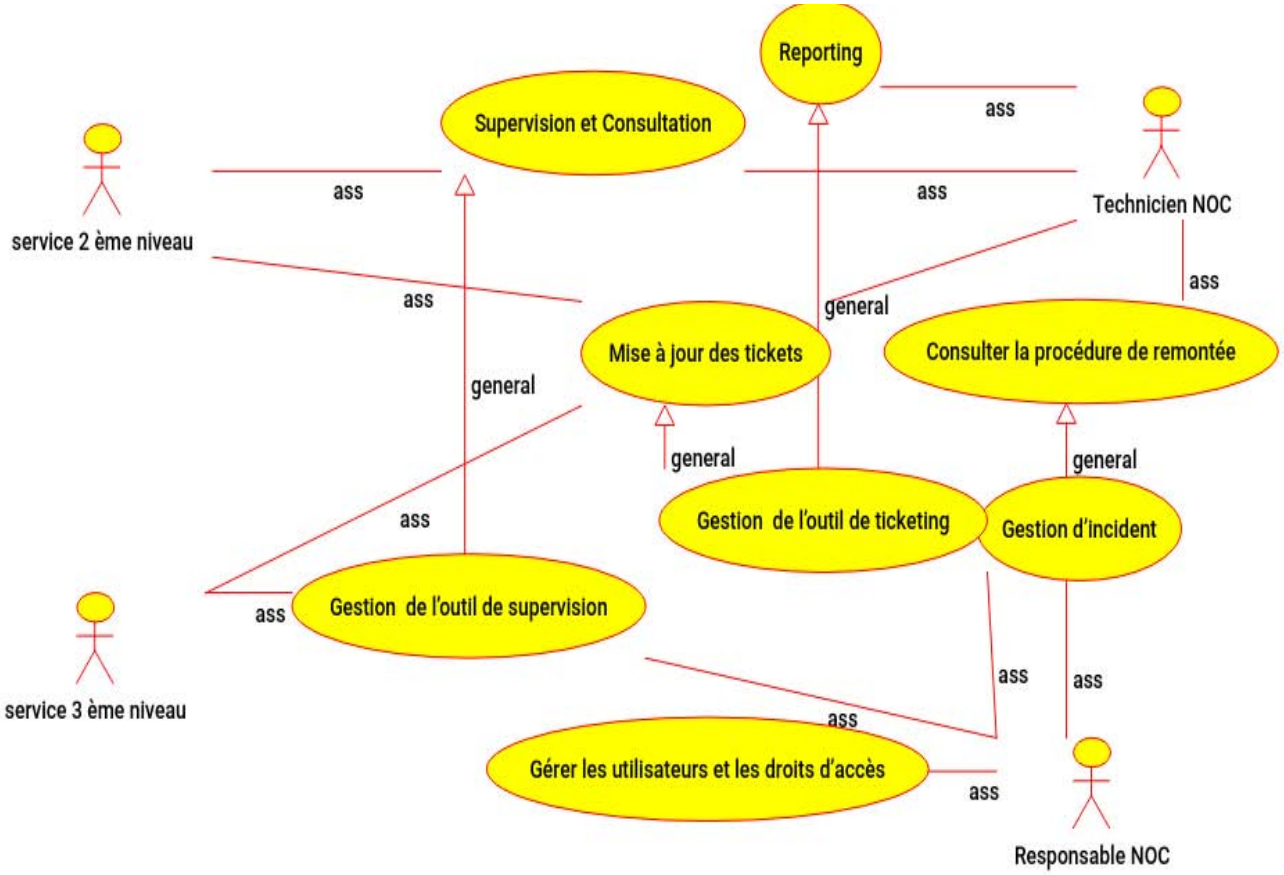


Figure 5. Cas d'utilisation d'un centre NOC.

Comme illustré dans la figure 5, ce cas d'utilisation englobe toutes les activités possibles dans le centre NOC, ainsi que les différents intervenants dans ce système.

Par la suite, nous allons découper le cas d'utilisation global sur les trois composants NOC : système de supervision, système ticketing et finalement système de gestion d'incident.

1.6.2.2 Cas d'utilisation système ticketing

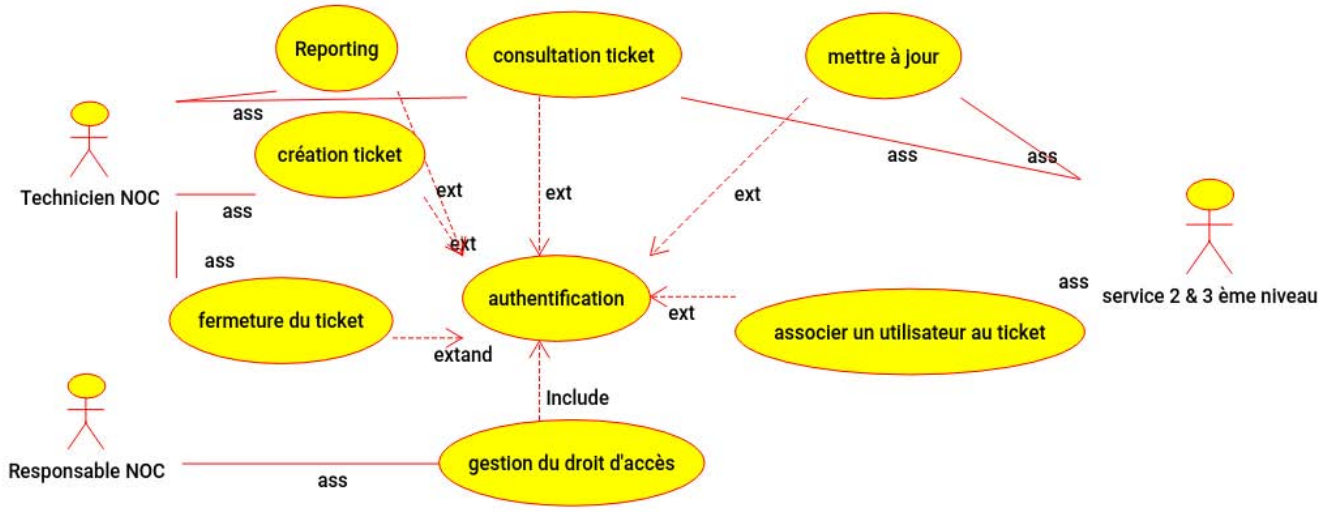


Figure 6. Cas d'utilisation système ticketing.

La figure 6 montre que le technicien NOC est l'acteur primordial de ce cas, il crée, met à jour et supprime les tickets suite à la détection d'un dysfonctionnement ou incident ; par ailleurs, les utilisateurs de 2ème et 3ème niveau jouent le même rôle dans ce système : ils peuvent consulter et mettre à jour les tickets qui lui sont assignés.

1.6.2.3 Cas d'utilisation système supervision

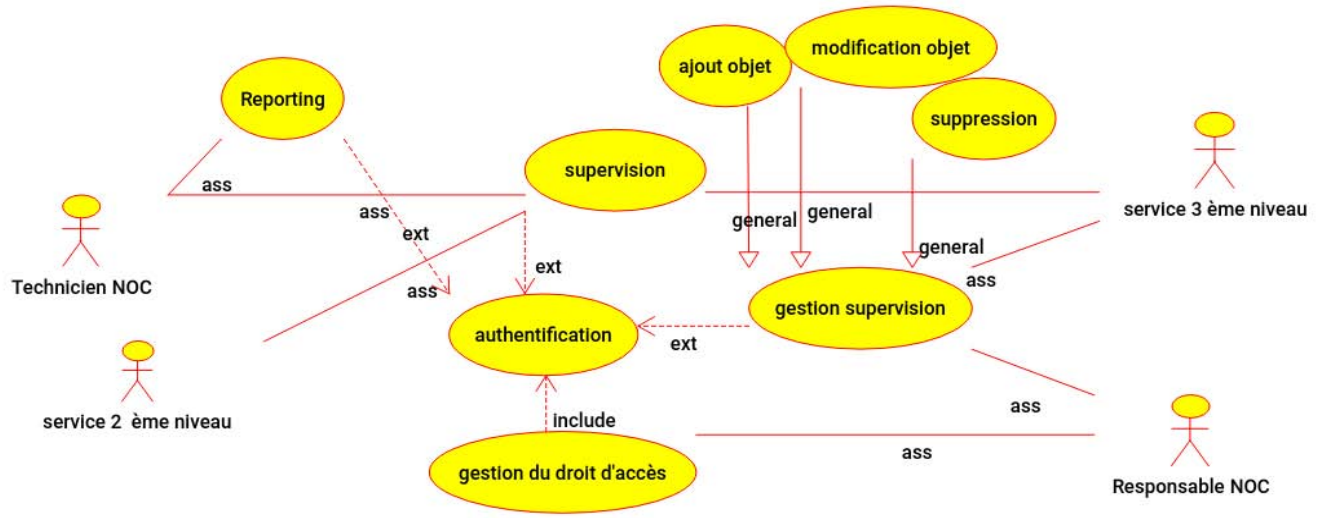


Figure 7. Cas d'utilisation système supervision.

Suite à une brève analyse de la figure 7, nous déduisons que la réalisation de ce cas d'utilisation se fait comme suit :

Le responsable NOC crée les profils et les utilisateurs ainsi que leurs droits. Il ajoute aussi les hôtes et les objets à surveiller. Ensuite, les autres utilisateurs peuvent superviser et consulter les objets dont ils ont le droit de visualiser.
Le technicien NOC peut générer ensuite le tableau de bord et en réaliser un rapport d'activité.

1.6.2.4 Cas d'utilisation gestion d'incidents

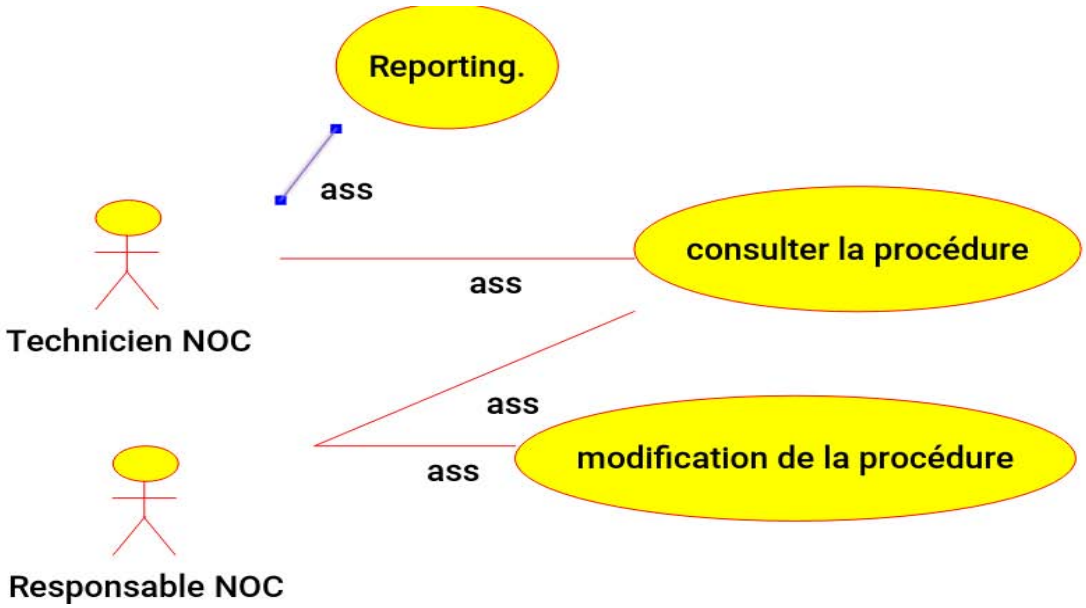


Figure 8. Cas d'utilisation gestion d'incidents.

La figure 8 montre que la gestion d'incident est un système interne au service NOC, il concerne uniquement l'équipe NOC : les techniciens NOC et le responsable.

1.7 Conclusion

Ce chapitre nous a permis, en premier lieu, d'introduire notre projet et de présenter l'état actuel du service monitoring ainsi que les attentes de l'entreprise d'accueil par la mise en place d'un centre NOC. En deuxième lieu, nous avons spécifié les besoins fonctionnels et non fonctionnels du système résultant de notre système, ce qui a permis de bien encadrer les rôles et les tâches des intervenants dans le système NOC. Le chapitre suivant présentera une étude approfondie de l'architecture d'un centre NOC.

Chapitre 2
Architecture d'un centre NOC

2.1 Introduction

Dans ce chapitre, nous allons exposer les outils que nous envisageons déployer afin de mettre en valeur les fonctionnalités qu'elles offrent permettant d'optimiser, enrichir et garantir la mise en place d'une solution complète, facile à administrer et qui répond aux besoins déjà fixés.

2.2 Architecture générale de la solution

La mise en place d'un centre NOC implique le déploiement et l'intégration de ces quatre modules essentiels (Figure 9) :

1. un outil de supervision,
2. un outil de ticketing qui assure à la fois la remontée et le reporting sur l'activité du centre NOC,
3. un outil de visualisation : intégré à l'outil supervision, il permet l'affichage des écrans paramétrés au mur d'écran (vidéo Wall),
4. un document qui structure la gestion de remontée.

Figure 9. Architecture NOC.

2.3 Composants de l'application

2.3.1 Outil de supervision

Parmi les produits de supervision connus sur le marché, on peut citer :

-chez les éditeurs : IBM Tivoli Monitoring, HP OpenView et BMC Patrol ;

-dans le monde libre : Nagios, ZABBIX, Cacti et Vigilo, Centreon,

Parmi les solutions libres, les deux logiciels Zabbix et Nagios sont les plus répandus et les plus utilisés. Par rapport à mon projet, ce sont les deux solutions les plus adaptées permettant de satisfaire pratiquement tous les besoins de l'entreprise, par les différentes fonctionnalités qu'elles offrent [7].

2.3.1.1 Comparaison des outils de supervision

Voici un tableau comparatif des deux logiciels choisis :

	Zabbix	**Nagios**
Présentation	-Multiplateformes. -Moteur en C, interface web utilisateur en PHP, base de données SQL (MySQL, Oracle…) -Configuration centralisée sur une même interface graphique. \ -Peut monitorer de 3 manières : _Lancement d'un processus sur les machines à monitorer pour collecter des données locales, grâce à l'agent Zabbix. _Requêtes SNMP. _Check externes qui sert à tester les services réseaux (tests limités à des pings ou test de protocoles).	-Conçu pour les plateformes Unix. -Moteur en C, perl, sharp…, interface web en PHP, base de données SQL. -Configuration plus ou moins complexe \ -Peut monitorer de 2 manières : _L'utilisation des journaux d'exploitation par l'envoie des évènements issus des fichiers log en temps réel vers un serveur centrale offrant les informations nécessaires à la supervision. _Supervision active des services et infrastructure qui nous permet de garder l'historique des performances.
Fonctionnalités	-Offre une interface web de consultation et d'administration. -Surveillance des ressources des serveurs (CPU, mémoire…). -générer des graphes et des écrans -Peut déclencher des alertes en envoyant des mails. -Supervise des équipements SNMP. -Gère les pannes et les performances -capable de faire du monitoring SNMP et IPMI ainsi que de la découverte de réseau	-Offre une interface web basée sur les CGL avec gestion des droits pour la consultation. -Génère des rapports de surveillance. -Surveillance des ressources des serveurs (CPU, mémoire…). -Surveillance des services réseaux.
Architecture	Architecture généralement basée sur : Serveur Zabbix, le cœur et moteur de	Architecture généralement basée sur : -Le moteur de l'application qui sert à

	l'application programmé en C. -Agents Zabbix pour la collection des informations locales. -Une interface web d'administration et consultation des données. -Une base de données SQL.	ordonnancer les tâches de supervision écrit en C. -Une interface web réalisée à l'aide des GCI (ground-controlled interception), décrivant la vue d'ensemble su système et les anomalies possibles. -Plusieurs plugins qui peuvent être complétés en fonction des besoins.
Avantages	-Multiplateforme. -Utilise peu de ressources -Configuration et utilisation aisée. -Interface vaste mais claire. -Richesse des sondes et tests possibles (supervision d'applications Web, par exemple). -Réalisation de graphiques, cartes ou screens. -Configuration par l'interface graphique. -Mise à jour de la configuration via l'interface Web de Zabbix. -Surveillances des sites web : temps de réponse, vitesse de transfert... -les agents Zabbix sont assez légers (écrits en C).	-Des plugins qui étendent les possibilités de Nagios. -Des plugins permettent aux utilisateurs de développer facilement ses propres vérifications de services. -La remontée des alertes est entièrement paramétrable grâce à l'utilisation de plugins (alerte par courrier électronique, SMS, etc…).
Inconvénients	-L'agent Zabbix communique les données en claire nécessité de sécuriser les données. -Peu d'interfaçage avec d'autres solutions commerciales.	-Difficile à installer et à configurer -Dispose d'une interface compliquée -Ne permet pas d'ajouter des hosts via Web -Besoin d'un autre outil comme CACTI pour faciliter sa configuration -Pas de représentations graphiques -Les mises à jour de la configuration se font en mode « lignes de commandes » et doivent être réalisées côté supervision comme côté serveur à superviser.

Tableau 2.1. Tableau comparatif Zabbix et NAGIOS [9].

3.3.1.2 Choix

En étant l'outil plus puissant de monitoring de conception moderne, facile à paramétrer, et aussi conçu pour les infrastructures larges, on retient Zabbix comme l'outil de supervision.

2.3.1.2 Architecture Plateforme Monitoring

Avec la version 2.0.2 de Zabbix utilisée, on a actuellement une architecture distribuée composée d'un serveur moteur, deux serveurs SQL répliqués et un serveur web.

La figure 10 représente l'architecture actuelle de la plateforme de monitoring :

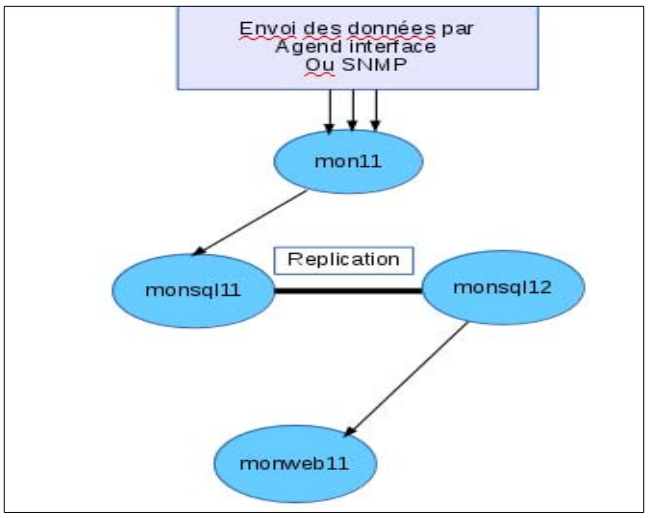


Figure 10. Architecture actuelle Monitoring.

Notre plateforme de supervision comprend quatre serveurs majeurs, dont les responsabilités sont décrites ci-dessous. Les agents Zabbix configurés au niveau de chaque serveur à surveiller.

- Serveur : mon11

Le serveur Zabbix est le composant central auquel les agents déclarent les informations et les statistiques de disponibilité et d'intégrité.

- Base de données MySQL (monsql11 et monsql12)

Toutes les informations de configuration ainsi que les données recueillies par mon11 sont stockées dans une base de données monsql11. Les données sauvegardées sont répliquées dans un autre serveur SQL monsql12. Ce serveur (monsql12) n'est plus utilisé à cause des multiples erreurs de réplication.

- Interface Web monweb11

Pour un accès facile à Zabbix à partir de n'importe où et de n'importe quelle plateforme, l'interface Web est fournie. L'interface s'exécute sur la même machine physique que celle qui exécute le serveur.

- Agent

Les agents Zabbix sont déployés afin de surveiller activement les ressources et les applications locales et de rapporter les données collectées au serveur Zabbix.

2.3.1.3 Architecture adoptée

On opte pour migrer la plateforme Monitoring en déployant les serveurs de version plus récente 3.x vers une architecture répliquée et distribuée sur deux PoP (points de présence) différents afin de garantir la disponibilité du service monitoring. Et notamment, même dans le cas d'incident qui touche tout un PoP, la supervision sera toujours garantie. La figure ci-dessous (figure 11) représente les six serveurs majeurs à déployer dans l'architecture future.

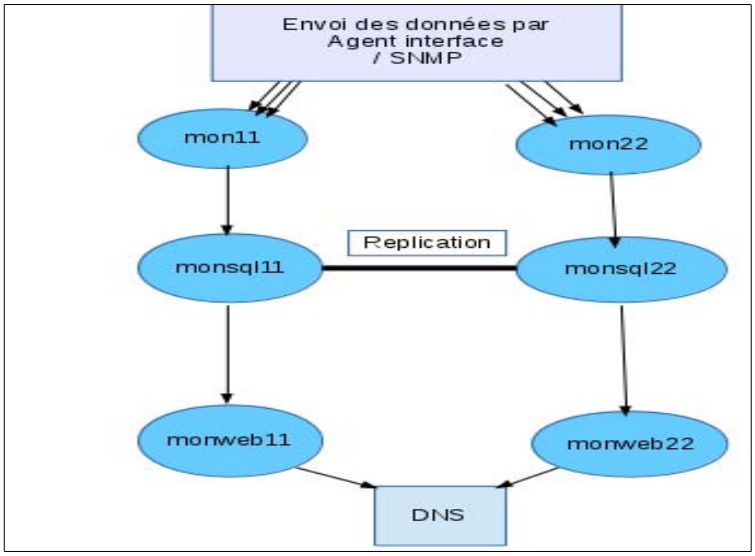


Figure 11. Architecture monitoring future.

2.3.2 Outil de ticketing

Parmi les logiciels de ticketing ou de suivi de problèmes, on se contente uniquement à ceux qui offrent plus de fonctionnalités en termes de gestion d'incident et la possibilité de générer des rapports d'activité. Donc, la comparaison des outils de ticketing sera entre Request Tracker et OTRS [10].

2.3.2.1 Comparaison des outils de ticketing

	Request Tracker	**OTRS**
Présentation	-une solution de ticketing éditée par Best Practical développée en perl	(Open source Ticket Request System) est une solution de gestion d'incidents
Fonctionnalités	-Création de tickets par mail, formulaire en ligne -Nombre illimité de projets	-Possibilité d'ouvrir un ticket soit via l'interface soit par mail -Personnalisation de l'interface via thème

	-Gestion d'historique -Gestion de la criticité -Possibilité de laisser des commentaires privés -Création de requêtes de recherches avec possibilité de sauvegarde de celles-ci. -Réponse automatique lors de l'ouverture d'un ticket, ou à la réception avec gestion de la personnalisation des mails via Template -Possibilité d'ajout de champs personnalisés -Gestion fines des permissions	-Possibilité d'intégration du SSO (Single Sign On) -Multi langage dont le Français Support du PGP pour les mails -Répondeur automatique pour les tickets ouverts par mail -Conversion automatique des mails HTLM en Text pour améliorer les recherches Notification par mail -Personnalisation de la vue des files d'attentes de tickets -Gestion d'historique des tickets -Gestion de priorités -Gestion du temps et de la facturation -Agent de création d'actions automatiques par actions planifiées -Fonctionnalité de Workflow -Calendrier avec temps de travail -Statistiques.

Tableau 2.2. Tableau comparatif RT et OTRS.

2.3.2.2 Choix

D'après cette brève comparaison, il est certes qu'OTRS répond le plus à notre besoin surtout concernant la gestion de priorités et la fonctionnalité de statistiques. De plus, il est à noter qu'OTRS est le plus efficace concernant la conformité ITIL et il permet aussi la gestion des changements, et configuration de la gestion des SLA.

2.3.3 Outil de visualisation au vidéo-wall

Kibana et Grafana sont deux outils open source qui permettent d'importer des données à partir de Zabbix et de les visualiser. Les deux outils pourraient être un intéressant complément de notre service de monitoring externe. Dans cette partie, nous allons dresser une comparaison entre ces deux outils afin de mettre en évidence les principales différences entre eux [12].

2.3.3.1 Comparaison des outils de visualisation

	Kibana	Grafana
Fonctionnalités	-Une analyse de données avancée. -Une visualisation des données dans une variété de types de tableaux, de tableaux et de cartes.	-Une présentation de graphes temporels basés sur des métriques spécifiques telles que l'utilisation des CPU et des E/S. -Grafana prend en charge de nombreux backends de

	-La création un tableau de bord log analytique complet.	stockage différents. -Grafana possède un éditeur de requêtes spécifique qui est personnalisé pour les fonctionnalités et les capacités qui sont incluses dans cette source de données [13].

Tableau 2.3. Tableau comparatif Kibana et Grafana.

2.3.3.2 Choix

Les deux outils sont de bonnes options et peuvent même parfois se compléter. L'outil Kibana peut être utilisé pour analyser les journaux de logs alors qu'on pourrait exporter les données dans Grafana comme une couche de visualisation. Nous allons opter pour Grafana afin d'enrichir notre vidéo Wall par les graphes et les écrans déjà configurés par l'outil Zabbix [14].

2.3.4 Document de gestion de remontée

Le document de gestion de remontée constitue un guide de mise en place d'un système de gestion d'incidents assurée par le service NOC. Il permet à l'ensemble des opérateurs NOC d'avoir un processus applicatif pour chaque évènement possible. Ainsi, il définit les éléments essentiels de la gestion des remontées : La détection, le diagnostic, la qualification, la procédure de remontée, la résolution et le reporting pour chaque type d'incident.

- **Types de remontée :**

Tout d'abord, il est préférable de lister tous les types d'incidents et alertes possibles afin de prévoir le traitement nécessaire :

- Comment détecter un problème ou un incident ?
- Quel est le diagnostic peut-on faire ?
- Quelle est sa catégorie et sa qualification ?
- À qui on remonte le ticket ?
- Comment écrire le rapport d'incident correspondant ?

- **Liste des types de remontée**

Ci-après, la liste des types de remontée possible :

-Plateforme Système

- Mail
- DNS
- Hosting
- Trafic serveurs

-Plateforme réseau

- Trafic Total

- Trafic par PoP

- Data Center FO

Ensuite, nous avons regroupé toutes les spécifications techniques que nous allons mesurées pour les déclenchements d'incidents et des alertes au niveau de chaque service. Aussi, nous avons identifié les types d'incidents et alertes possibles par son analyse, et spécifier la remontée et le reporting qui les concerne.

- **Processus de la gestion d'incident**

Le processus de la remontée est représenté par le schéma ci-dessous :

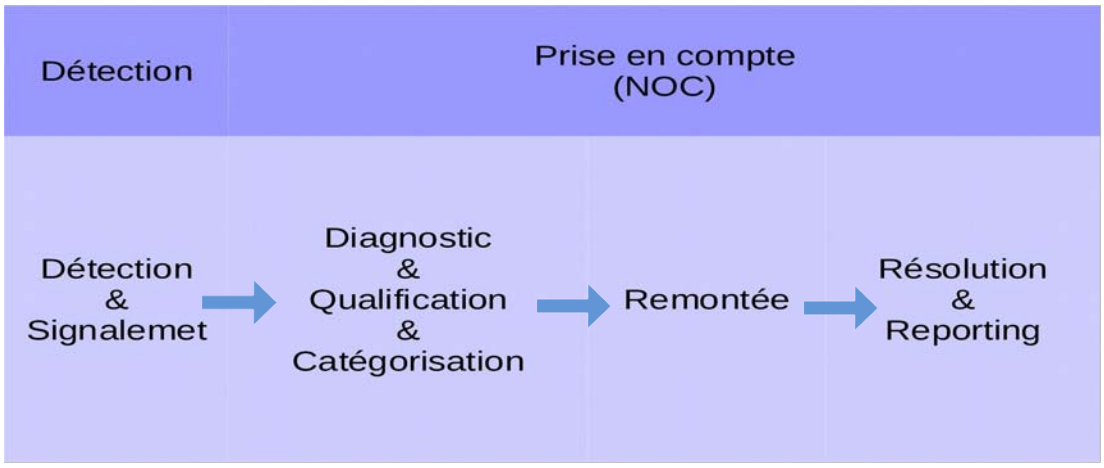


Figure 12. Processus de gestion des incidents.

Comme illustré dans la figure 12 le processus de gestion des incidents comporte les majeurs éléments suivant : la détection, le diagnostic, la qualification, la remontée, la résolution et le reporting.

- **Détection et diagnostic**

La détection et le signalement d'un événement susceptible d'être qualifié incident est réalisée soit par une personne ou par les moyens techniques (outils de supervision, gestion des logs,...).

La prise en compte est réalisée par l'équipe NOC à l'aide de l'ouverture d'un ticket pour que l'événement soit enregistré de manière à pouvoir en faire le diagnostic approprié.

- **Qualification**

Ensuite, l'équipe NOC sera amenée à transmettre l'incident au service concerné tout en précisant la priorité de l'incident (qualification).

Ce tableau décrit les quatre priorités à appliquer :

Priorité	Nom	Définition	Délai de résolution
P1	Majeure	Interruption complète d'un service, d'un système, du réseau, d'une application ou d'un élément de configuration identifié comme critique.	1 heure
P2	Élevée	S'applique lorsque le service, le système, le réseau, l'application ou l'élément de configuration peut procéder mais dont la performance est considérablement réduite et/ou les fonctionnalités sont très limitées	4 heures
P3	Normale	Un événement qui provoque la perte minime d'un service. Une solution permanente ou de contournement est disponible pour restaurer la fonctionnalité du service.	12 heures
P4	Basse	Un événement constituant un dérangement pour l'utilisateur, pour lequel il existe une alternative ou une réparation possible mais qui n'empêche en rien l'utilisateur de travailler	1 jour

Tableau 2.4. Tableau des priorités.

Ensuite, le ticket sera immédiatement soumis à l'équipe de réponse adéquate (Niveau 2).

Cette équipe sera amenée à répondre au ticket. La réponse doit comporter un traitement détaillé et des actions correctives réalisées afin de résoudre l'incident et d'autres planifiées.

- **Remontée**

Voici un schéma d'escalade d'un incident sur les trois niveaux de remontée, à commencer par le centre d'opération réseau :

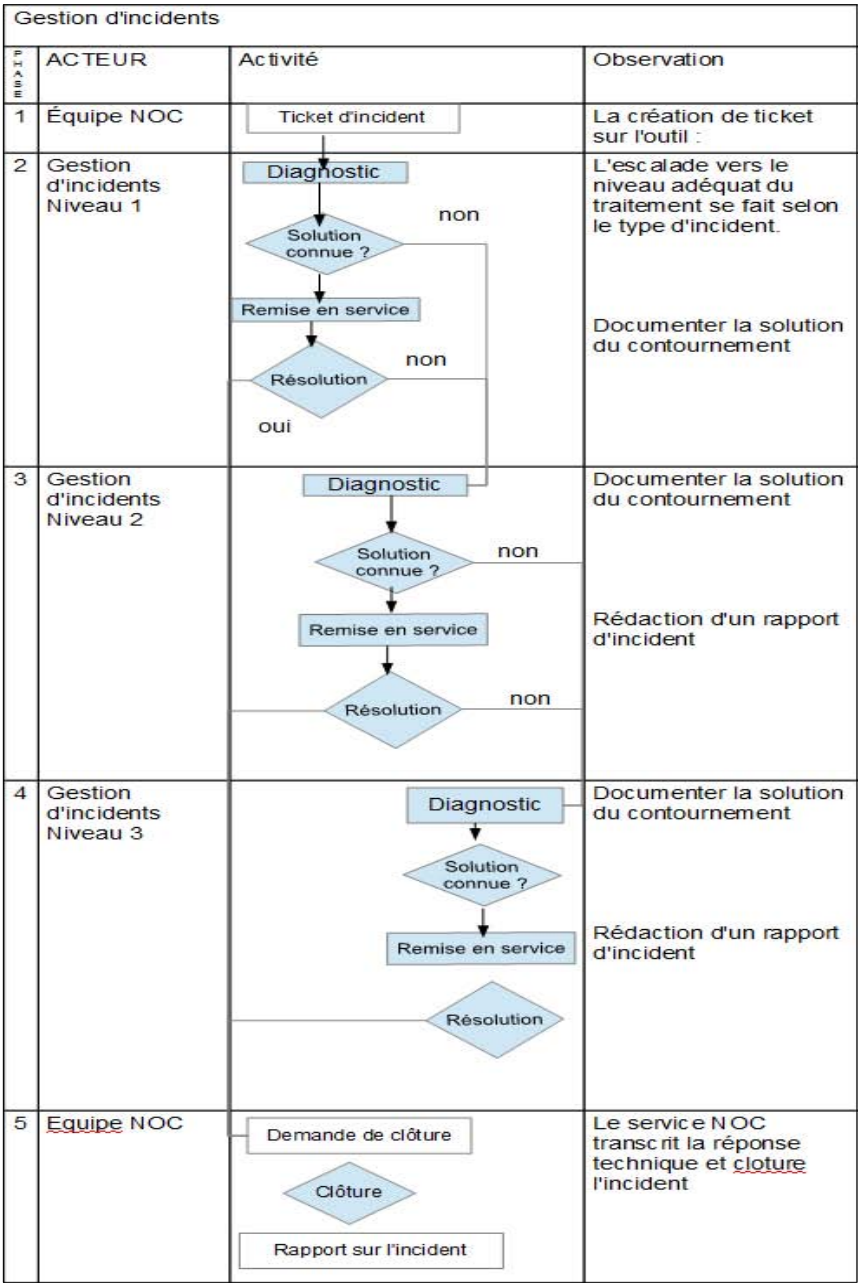


Figure 13. Schéma de remontée.

L'escalade est prévue dans le processus afin de transférer un incident du niveau 2 au niveau supérieur au bout du dépassement de la durée maximale de la résolution ou de la gravité de l'incident.

◆ **Résolution**

Le service NOC est le propriétaire de l'incident et en est responsable du suivi jusqu'à la résolution et sa fermeture.

◆ **Reporting**

Chaque incident doit être accompagné d'un rapport de synthèse. Ce rapport est rédigé par l'équipe NOC. Il doit présenter le traitement de résolution de l'incident et les actions correctives prévues. Il doit être enregistré dans l'outil de gestion d'incidents servant de base de connaissance aux incidents. Cette base d'information pourra par la suite être mise en profit pour une meilleure anticipation d'incidents, pour définir les contres mesures les plus appropriés ou encore vérifier si les actions correctives actées ont été suivies d'effets.

2.4 Conclusion

Le but de ce chapitre était de présenter les compléments que nous avons choisis au centre NOC. Certains ont été choisis pour leur nécessité comme Zabbix et la procédure de remontée, et d'autres participaient surtout à l'amélioration de la visualisation, et surtout à la facilité de sa configuration.

Le chapitre suivant entamera l'aspect technique de mon projet, de la mise en place jusqu'aux exemples d'utilisations.

Chapitre 3
Mise en place NOC

3.1 Introduction

Dans ce chapitre, nous allons détailler le déploiement de la solution NOC. Dans un premier temps, nous allons présenter l'environnement de travail de l'outil Zabbix, l'outil OTRS ainsi que l'outil Grafana. Dans un deuxième temps, nous allons décrire les différentes étapes de l'installation des trois composants de la solution.

3.2 Environnement de travail

3.2.1 Installation de l'environnement Zabbix

En se basant sur les recommandations de Zabbix (documentation –section prérequis [17]) ainsi que la taille de l'environnement à surveiller (entre 200 et 300 équipements), nous avons préparé trois machines virtuelles avec les caractéristiques suivantes :
Zabbix version : 3.2.4

	ZabbixServer	ZabbixSQL	ZabbixWeb
CPU	2 CPU	4 CPU	4 CPU
RAM	2 Go	4 Go	4 Go
Disque	40 Go	80 Go	40 Go
Sys exploitation	Centos 7	Centos 7	Centos 7

Les étapes de l'installation du système d'exploitation en annexes [15] [16].

L'architecture distribuée optée se compose de six serveurs hébergés sur deux points de présence distincts. A ce stade de travail, nous avons déployé uniquement trois serveurs pour la phase transitoire (comme présenté dans la figure 14).

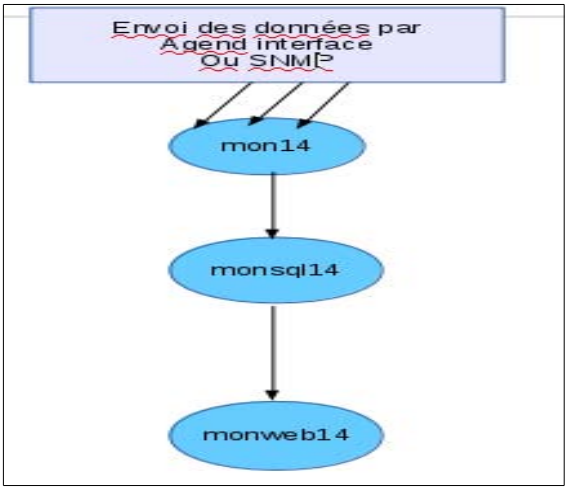


Figure 14. Architecture Zabbix déployée.

Nous avons opté pour une installation minimale et en mode texte pour alléger les serveurs et pour des raisons de sécurité (le mode graphique présente toujours des vulnérabilités et il n'est pas nécessaire).

3.2.2 Installation de l'environnement OTRS

Nous avons opté pour une installation minimale de ressources par défaut.
Version OTRS 5

	OTRS
CPU	2 processors
RAM	2048+1024 M
Disque	2 Go
sys exploitation	Opensuse Leap 42.2

Les étapes de l'installation du système d'exploitation sont présentées en annexes.

3.2.3 Installation de l'environnement Grafana

Nous avons opté pour une installation minimale de ressources par défaut.

Version **Grafana**5

	Grafana
CPU	1 processor
RAM	4 Go
Disque	20 Go
Sys exploitation	Centos 7

Les étapes de l'installation du système d'exploitation sont en annexes.

3.3 Etapes d'installation

Dans cette partie, nous présentons, en détail, les étapes de l'installation des trois composants NOC : Zabbix, OTRS et Grafana.

3.3.1 Etapes d'installation Zabbix

Voici les étapes d'installions du moteur Zabbix (mon14), serveur SQL (monsql14) et serveur web Zabbix (monweb14) :

mon14

wget https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
rpm -ivh mysql57-community-release-el7-9.noarch.rpm
yum install mysql
yum -y install
"http://repo.zabbix.com/zabbix/3.2/rhel/7/x86_64/zabbix-release-3.2-1.el7.noarch.rpm"
yum clean all
yum -y install zabbix-agent zabbix-get zabbix-server-mysql
systemctl status zabbix-server
systemctl stop zabbix-server
systemctl start zabbix-server

monsql14

wget https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
rpm -ivh mysql57-community-release-el7-9.noarch.rpm
yum install mysql-server
yum -y install
"http://repo.zabbix.com/zabbix/3.2/rhel/7/x86_64/zabbix-release-3.2-1.el7.noarch.rpm"
yum clean all
yum -y install zabbix-agent

monweb14

wget https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
rpm -ivh mysql57-community-release-el7-9.noarch.rpm
yum install mysql
yum -y install
"http://repo.zabbix.com/zabbix/3.2/rhel/7/x86_64/zabbix-release-3.2-1.el7.noarch.rpm"
yum clean all
yum -y install zabbix-web-mysql zabbix-agent
systemctl start httpd

Après avoir redémarré le service apache, il faut se connecter via le navigateur web sur cette adresse http : http://Ip/zabbix pour terminer les étapes d'installation web : (voir figure 15)

Figure 15. Installation Web Zabbix.

3.3.2 Etapes d'installation OTRS

Ci-dessous les étapes d'installation de l'outil de ticketing OTRS [22] :

2017-03-24 09:48:51 zypper install mysql perl-DBD-mysql
2017-03-24 09:49:43 vim /etc/my.cnf
2017-03-24 09:53:54 rcmysql restart
2017-03-24 09:55:30 /usr/bin/mysql_secure_installation
2017-03-24 09:57:34 chkconfig -a mysql
2017-03-24 12:46:42 wget -r 'http://ftp.otrs.org/pub/otrs/RPMS/suse/11/otrs-5.0.17-01.noarch.rpm'
2017-03-24 12:47:37 rpm -Uvh otrs-5.0.17-01.noarch.rpm
2017-03-27 14:41:15 zypper update
2017-03-27 14:43:28 sudo /opt/otrs/bin/otrs.CheckModules.pl
2017-03-27 14:43:58 zypper install perl-Crypt-Eksblowfish
2017-03-27 14:44:35 zypper install perl-DBD-Pg
2017-03-27 14:44:56 zypper install perl-Encode-HanExtra
2017-03-27 14:45:35 zypper install perl-JSON-XS
2017-03-27 14:49:10 zypper install perl-Mail-IMAPClient
2017-03-27 14:49:26 sudo /opt/otrs/bin/otrs.CheckModules.pl
2017-03-27 14:49:32 /opt/otrs/bin/otrs.CheckModules.pl
2017-03-27 14:49:46 zypper install perl-Template-Toolkit
2017-03-27 14:50:19 zypper install perl-Text-CSV_XS
2017-03-27 14:50:43 zypper install perl-XML-LibXSLT
2017-03-27 14:50:59 zypper install perl-YAML-LibYAML
2017-03-27 14:54:12 /opt/otrs/bin/otrs.CheckModules.pl

2017-03-27 14:54:29 zypper install perl-XML-LibXSLT
2017-03-27 14:56:05 perl -MCPAN -e 'install DBD::Pg'
2017-03-27 14:59:17 /opt/otrs/bin/otrs.CheckModules.pl
2017-03-27 15:04:55 /usr/sbin/rcapache2 status
2017-03-27 15:15:27 /opt/otrs/bin/otrs.CheckModules.pl
2017-03-27 08:20:45 rcapache2 restart
2017-03-27 08:21:23 chkconfig -a apache2
2017-03-28 08:44:21 chmod +x /opt/otrs/bin/otrs.Daemon.pl
2017-03-28 08:44:29 /opt/otrs/bin/otrs.Daemon.pl start
2017-03-28 08:48:53 vi /opt/otrs/bin/otrs.Daemon.pl
2017-03-28 08:49:17 vi /opt/otrs/bin/otrs.SetPermissions.pl
2017-03-29 11:14:55 bin/otrs.Daemon.pl status
2017-03-29 11:15:05 ssh otrs@172.16.3.175
2017-03-29 11:17:43 bin/otrs.Daemon.pl status
2017-03-29 11:17:47 ssh otrs@172.16.3.175
2017-03-29 16:29:58 /etc/init.d/apache2 restart

Ensuite, l'installation web OTRS se poursuit sur cette URL : http ://ip/otrs/intaller.pl [23] (voir figure.16)

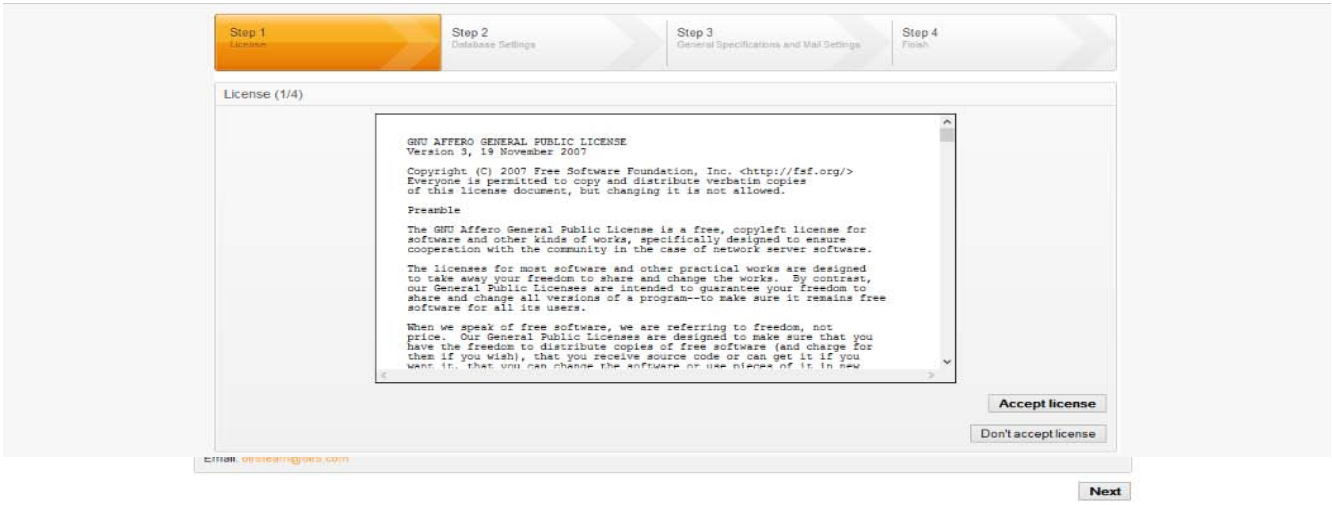


Figure16.Installation Web OTRS.

3.3.3 Etapes d'installation Grafana

Ci-dessous les étapes d'installation de l'outil de visualisation Grafana :

2017-05-05 15:21:28 wget https://s3-us-west-2.amazonaws.com/grafana-releases/release/grafana-4.2.0-1.x86_64.rpm
2017-05-05 15:25:36 vim /etc/yum.repos.d/grafana.repo

2017-05-05 15:27:09 sudo yum install grafana
2017-05-05 15:27:38 sudo service grafana-server start
2017-05-05 15:27:48 sudo /sbin/chkconfig --add grafana-server
2017-05-05 15:28:20 systemctl start grafana-server
2017-05-05 15:28:28 sudo systemctl enable grafana-server.service

Ci-dessous est une capture d'écran du terminal au niveau du serveur lors de l'installation de Grafana (voir figure 17) :

Figure 17. Capture d'écran lors de l'installation de Grafana.

Intégration Zabbix Grafana

Afin de configurer Grafana pour accéder à la base de données de Zabbix, voici la démarche que nous avons suivie [25] :

2017-05-05 15:52:08 sudo grafana-cli plugins install alexanderzobnin-zabbix-app

2017-05-05 15:53:12 sudo systesmctl restart grafana-server

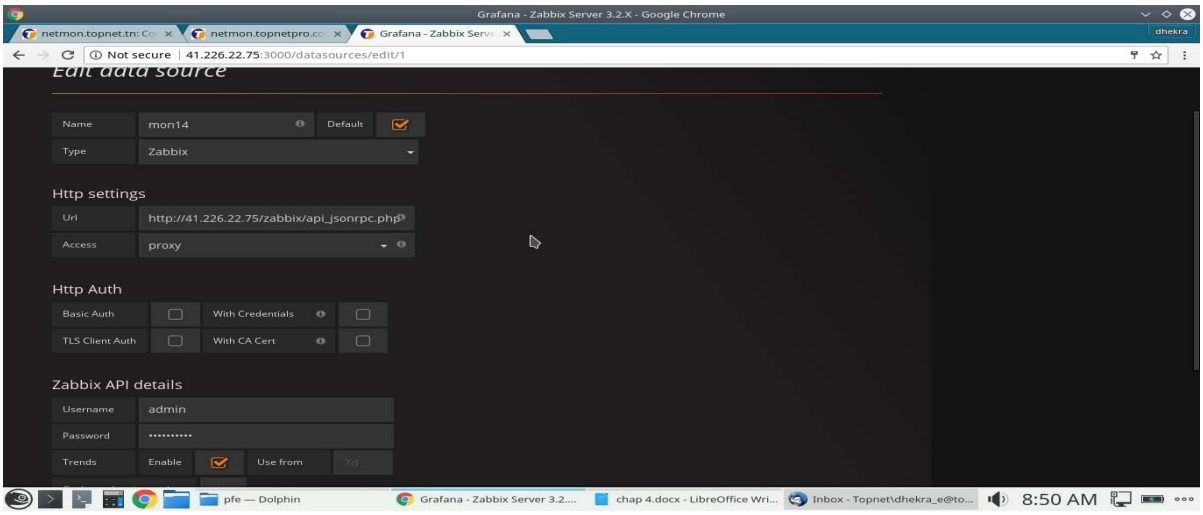


Figure 18. Intégration Zabbix Grafana.

La figure 18 représente la configuration paramétrée au niveau de l'outil Grafana qui lui permet la collecte des données depuis l'outil Zabbix.

3.4 Interface application

3.4.1 Interface Zabbix

Après avoir installé l'outil Zabbix, notre travail consiste à l'implémentation de l'outil :

- ajout des hôtes à surveiller (plus que 300 hôtes).

- ajout des templates et leur association aux hôtes selon les spécifications techniques à mesurer.

- création des graphes et écrans de monitoring.

-personnalisation de l'interface Zabbix.

-ajout des utilisateurs selon leur droit d'accès à l'outil.

Le tableau de bord global est l'élément central de l'interface web. Il fournit des détails sur l'environnement surveillé. Les informations décrites par la figure 19 sont disponibles sur un seul écran :

-État du serveur Zabbix.
-État du système.
-État de l'hôte.

-20 derniers problèmes.-Surveillance Web.-Etat de la découverte.
-Lien vers les graphiques préférés.

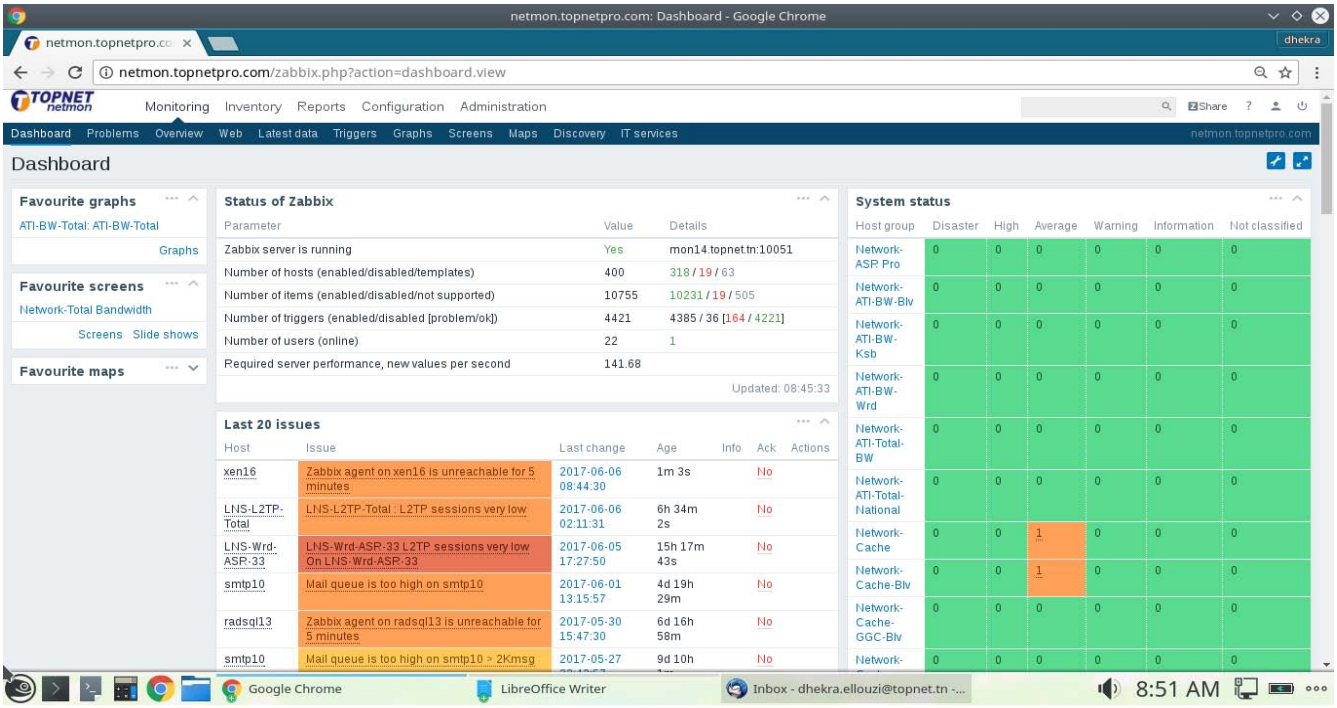


Figure 19. Tableau de bord de l'outil Zabbix.

L'ancienne plateforme de monitoring présente un problème de ressources. Ce qui a provoqué une lenteur de chargement et des pannes répétitives. Grâce à la migration vers la version 3.2, plusieurs avantages et nouvelles fonctionnalités seront disponibles :

-Une interface utilisateur remodelée dans les standards du web actuel (voir Fig. 19).

-Le chiffrement des échanges basé sur le protocole TLS (Transport Layer Security).

-Les fonctions « prévision » et « temps restant » qui permettent de savoir dans combien de temps vous allez passer en dessous du seuil critique ou de savoir où vous en serez de votre espace disque dans 1 heure.

-La possibilité de partager des écrans et diaporamas avec un utilisateur ou un groupe d'utilisateurs.

-L'authentification SMTP.

-La création de dépendances entre prototypes de déclencheurs (prototypes triggers).

-La possibilité de Grapher des éléments de type log.

…

Figure 20. Écran de l'outil Zabbix dans les deux versions.

Comme indique la figure 20, même au niveau des graphes et des écrans, nous avons remarqué une amélioration de performances, de précision et une facilité d'utilisation.

3.4.2 Interface OTRS

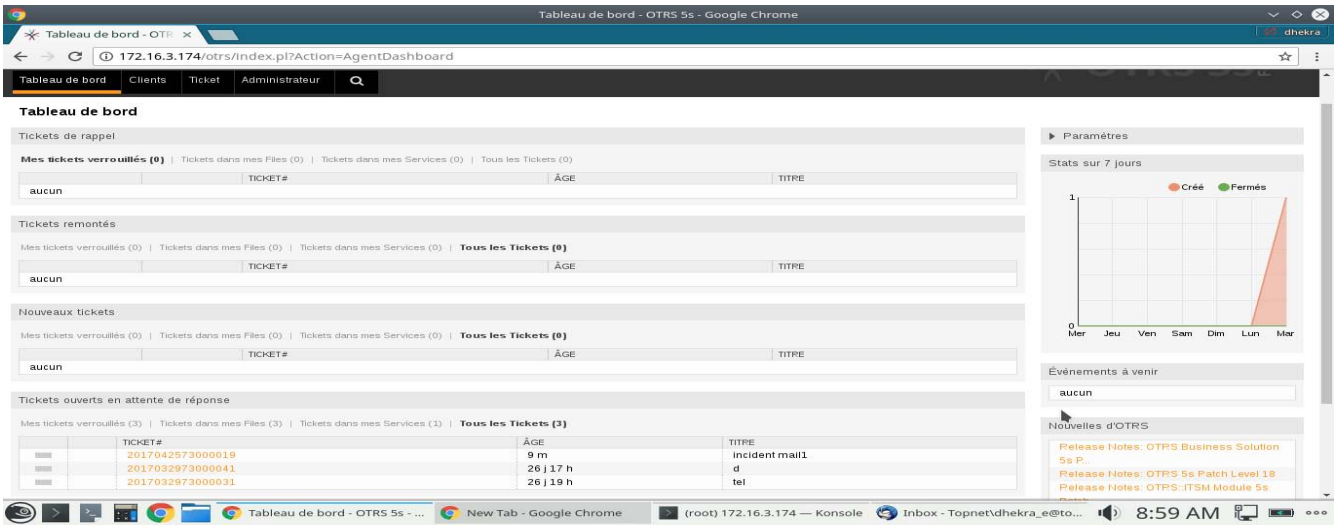


Figure 21. Tableau de bord d'OTRS.

La figure 21 représente le tableau de bord OTRS. Ce dernier est personnalisé selon l'utilisateur, il fournit une vue globale sur les tickets assignées et l'historique des tickets traitées.

3.4.3 Interface Grafana

Après avoir installé et intégré Grafana à Zabbix, nous avons ensuite créé de nouveaux écrans de visualisation. Par exemple, la figure 22 représente l'état du serveur Zabbix : CPU, trafic réseau, les derniers alertes, mémoire disponible,…

Figure 22. Écran de visualisation Grafana.

3.5 Exemple de Remontée

Avant la mise en place de l'outil OTRS, la remontée n'été pas structurée ; les pannes et incidents sont détectés soit par l'outil Zabbix, soit par la réception d'un email du service support sans passer par le diagnostic nécessaire.

Nous présenterons, dans cette partie deux cas de remontées possibles :

3.5.1 Exemple 1 : Observation d'une queue pour un serveur SMTP

Impact sur les clients : lenteur d'envoi des emails

Priorité : P2

-Diagnostic effectué avant la remontée :

-Observation des graphes des serveurs de même fonctionnalité.

-test d'envoi depuis une @mail externe vers une @ mail Topnet (délai de réception du mail).

-Telnet port SMTP 25.

-test d'envoi d'un email depuis le serveur concerné.

-vérification du Top Sender.

-Vérification des performances des hyperviseurs qui hébergent le serveur.

-Vérification de la queue pour ce serveur.

-Création d'un ticket par email pour la remontée et affectation du ticket au service mail

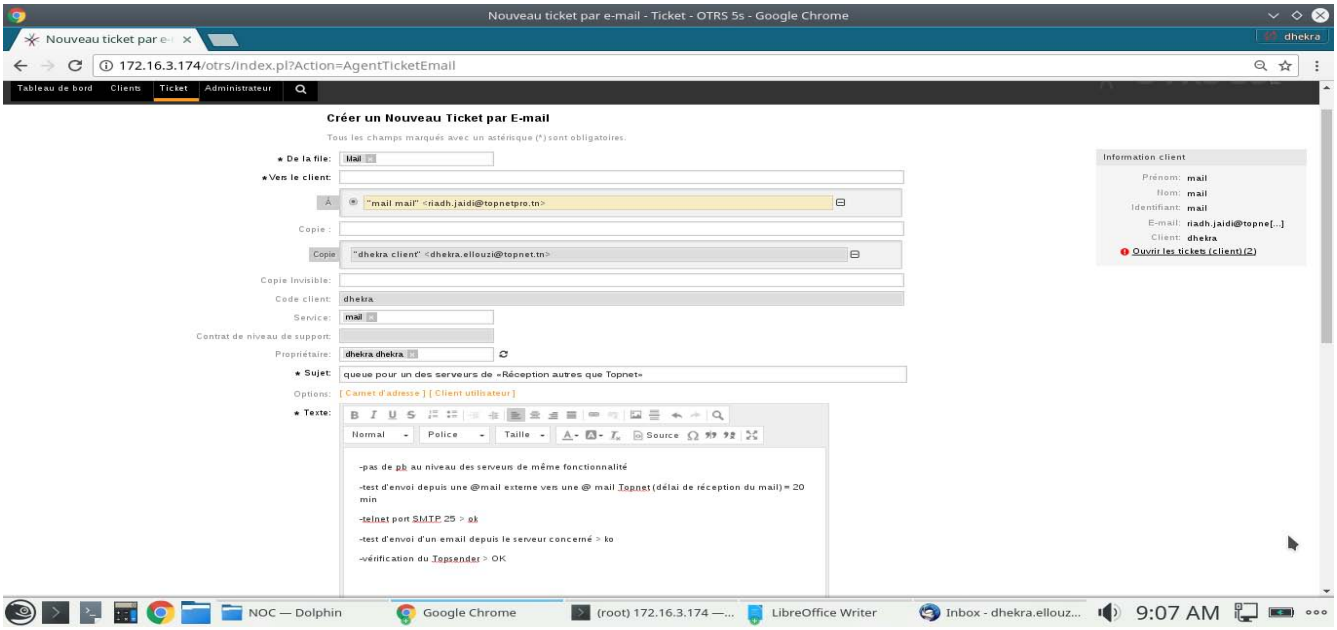


Figure 23. Création du ticket sur l'outil OTRS.

L'équipe mail reçoit un email et doit répondre à ce ticket dans un délai de 4h. Après ce délai, un email de notification sera envoyé au responsable hiérarchique.

Priorité	Délai de remontée au niveau 3
P1	1 heure
P2	4 heures
P3	12 heures
P4	1 jour

-Résolution

Après la réception d'une réponse du service mail, l'équipe NOC doit effectuer une vérification de la résolution de l'incident avant la clôture du ticket. Un rapport d'incident doit être rédigé et sauvegardé au niveau de l'outil de la gestion d'incident.

3.5.2 Exemple 2 : Accessibilité Zabbix agent : au niveau d'un des Web Servers

Impact sur les clients : sites web hébergés sur cette machine inaccessible

Priorité : P1

-Diagnostic effectué avant la remontée :

-Ping @ machine.

-Vérification des autres machines hébergées dans le même hyperviseur.

-Observation de l'état pendant 10 minutes avant la remontée.

-Création d'un ticket par email pour la remontée et affectation du ticket au service Hosting

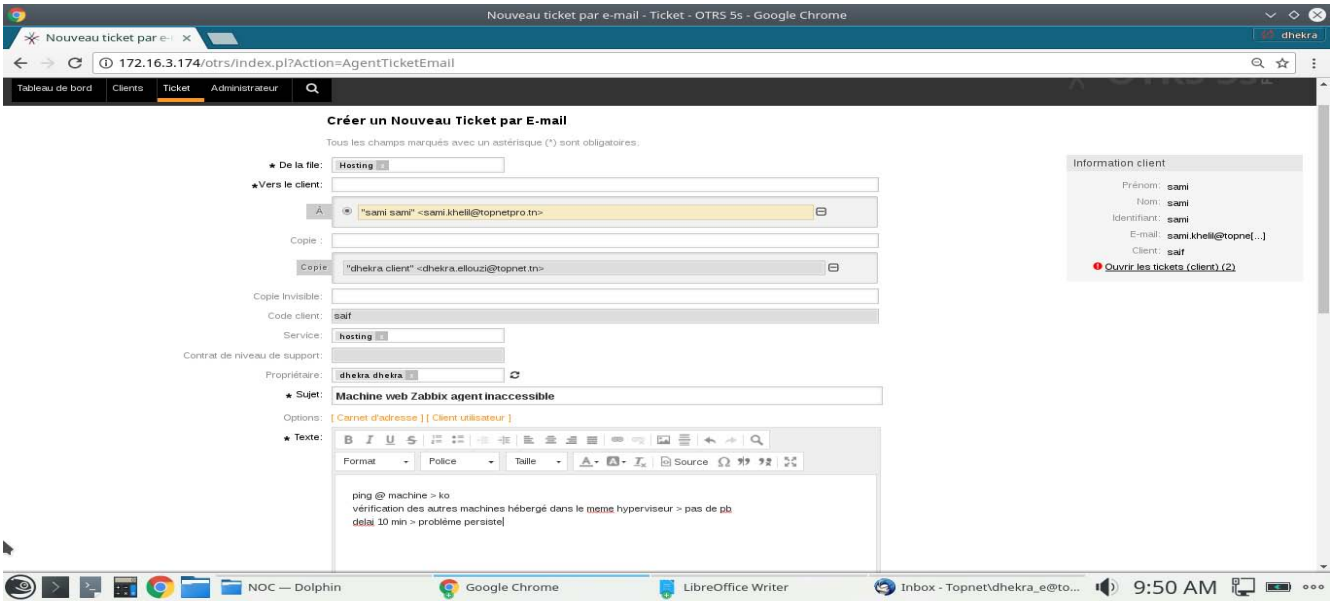


Figure 24. 2ème Exemple de remontée.

L'équipe Hosting reçoit un email et doit répondre à ce ticket dans un délai de 1h. Après ce délai, un email de notification sera envoyé au responsable hiérarchique.

-Résolution

Après la réception d'une réponse du service Hosting, l'équipe NOC doit effectuer une vérification de la résolution de l'incident avant la clôture du ticket. Un rapport d'incident doit être rédigé et sauvegardé au niveau de l'outil de la gestion d'incident.

3.6 Conclusion

Dans ce chapitre nous avons présenté les étapes de la mise en place et l'utilisation de notre solution. Par la suite, nous avons exposé deux scénarios de remontées d'incidents.

Conclusion Générale

Le domaine de la supervision est un domaine important de l'administration systèmes et réseaux. En constante évolution, les solutions de supervision ont prouvé qu'elles avaient leur place dans la sphère professionnelle.

Comme nous avons déjà explicité dans ce projet, la supervision et la gestion d'incidents sont deux moyens indispensables pour favoriser la continuité des services Internet. Le but de ce projet était de choisir une solution complète qui répondait aux besoins de l'entreprise en mettant en place un centre NOC.

D'une part, l'association des outils Zabbix et Grafana a permis la constitution d'une solution de monitoring à la fois puissante et efficace offrant un vidéo-wall de monitoring global et des écrans de monitoring bien définis. D'autre part, le document de gestion d'incident consiste à une référence de remontée pour les utilisateurs de l'application OTRS.

Pour conclure, nous estimons avoir satisfait les objectifs initialement fixés, mais comme tout œuvre humaine, cette application pourrait être rectifiée au fur et à mesure de son utilisation. On peut aussi l'améliorer par :

-l'ajout de la notification par SMS et TWILIO à l'outil Zabbix.

-la mise en place d'une autre plateforme de monitoring dans un autre PoP répliquée à la plateforme existante afin d'assurer la haute disponibilité de l'outil de supervision.

-l'utilisation de l'escalade SLA et l'ajout d'autres rapports journaliers à l'outil OTRS.

-la mise à jour régulière du document de gestion d'incidents.

Webographie

[1] « Centre d'opérations du réseau ». [En ligne]. Disponible sur : https://fr.wikipedia.org/wiki/Centre_d%27op%C3%A9rations_du_r%C3%A9seau.

[2]http://www.sikich.com/images/technology/NOC.jpg[Enligne].Disponible sur :

http://www.sikich.com/it/technology-consulting/network.

[3] « Why Your SOC and NOC Should Run Together but Separately ». [En ligne]. Disponible sur :https://ayehu.com/why-your-soc-and-noc-should-run-together-but-separately/

[4] Alain Aina « La gestion réseau et le NOC: concepts, pratiques, et outils ». [En ligne]. Disponible sur :

[5] « ITIL V2 La gestion des incidents ». [En ligne]. Disponible

sur http://www.itilfrance.com/pages/docs/hgelun/itilv2_incidents.pdf

[6] charleskafami « Cyber Security: Security Operations Center (SOC) vs. Network Operations Center (NOC) ». [En ligne]. Disponible sur :http://www.intellectualpoint.com/blog/?p=320

[7] « Zabbix, le successeur de Nagios ? ». [En ligne]. Disponible sur : http://blog.neoxia.com/zabbix/

[8] Mamadou Bassirou ARRY « Comparaison des outils de supervison ». [En ligne]. Disponible sur : https://www.supinfo.com/articles/single/3124-comparaison-outils-supervison

[9] Othman Souli « Mise en place d'un système de supervision Open source. ». [En ligne]. Disponible sur : http://pf-mh.uvt.rnu.tn/573/1/Mise_en_place_d%E2%80%99un_syst %C3%A8me_de_supervision_Open_source..pdf

[10] Pierre-Yves Dubreucq « Solution de HelpDesk – Gestion d'incidents Opensource». [En ligne]. Disponible sur : http://blog.admin-linux.org/logiciels-libres/solution-de-helpdesk-gestion-incident-opensource

[11] Wahid Mejri & Mohamed Slim Arafa « Conception et réalisation d'une application de gestion des comptes mail et internet». [En ligne]. Disponible sur : http://pf-mh.uvt.rnu.tn/578/1/Conception_et_r %C3%A9alisation_d%E2%80%99une_application_de_gestion_des_comptes_mail_et_internet.pdf

[12] «Kibana vs. Grafana vs. Graphite Get help choosing one of these ». [En ligne]. Disponible sur : https://stackshare.io/stackups/graphite-vs-grafana-vs-kibana

[13] Olivier Jan « Grafana : L'interface web de Graphite ». [En ligne]. Disponible sur : https://wooster.checkmy.ws/2014/03/grafana-graphite-interface/

[14] Asaf Yigal « Grafana vs. Kibana: The Key Differences to Know ». [En ligne]. Disponible sur : https://logz.io/blog/grafana-vs-kibana/

[15] Vadym Kalsin« How To Install and Configure Zabbix to Securely Monitor Remote Servers on CentOS 7 ». [En ligne]. Disponible sur : https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-zabbix-to-securely-monitor-remote-servers-on-centos-7

[16] Rahul K. , « How To Install Zabbix Server 3.0 on CentOS/RHEL 7/6/5 ». [En ligne]. Disponible sur : https://tecadmin.net/install-zabbix-network-monitoring-on-centos-rhel-and-fedora/#

[17] « Zabbix Documentation 3.0». [En ligne]. Disponible sur : https://www.zabbix.com/documentation/3.0/manual/installation/requirements

[18] Alain Ganuchaud « Optimisation de la plateforme de Supervision Zabbix ». [En ligne]. Disponible sur : https://fr.slideshare.net/AlainGanuchaud/zabbix-performancetuningfr-v3

[19] « Install Zabbix 3.0 (Monitoring Server) on CentOS 7.x / RHEL 7.x ». [En ligne]. Disponible sur : https://www.linuxtechi.com/install-zabbix-3-0-on-centos-7-rhel-7/

[20] Matei Cezar « Setting Up “NTP (Network Time Protocol) Server” in RHEL/CentOS 7 ». [En ligne]. Disponible sur : https://www.tecmint.com/install-ntp-server-in-centos/

[21] « monitorando-um-equipamento-via-zabbix-com-simple-check ». [En ligne]. Disponible sur : http://www.proger.eti.br/2009/10/monitorando-um-equipamento-via-zabbix-com-simple-check/

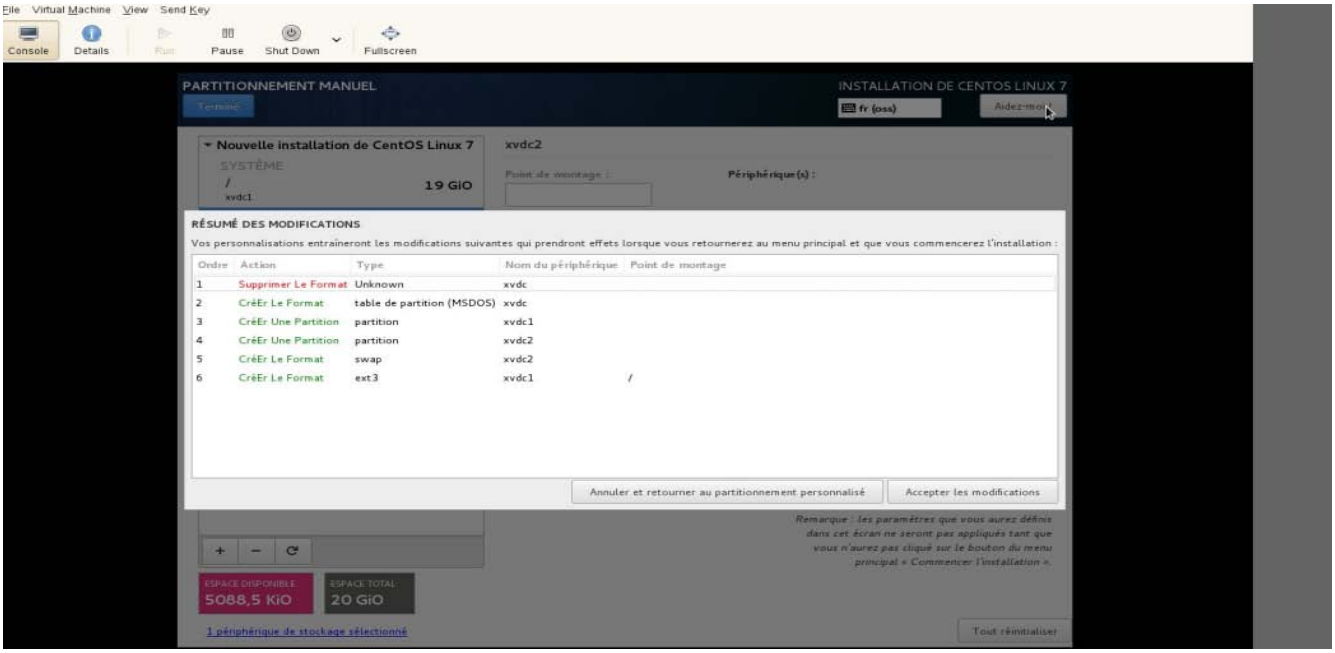
[22] «OTRS Portal - Chapter 2. Installation ». [En ligne]. Disponible sur : http://doc.otrs.com/doc/manual/admin/3.3/en/html/installation.html

[23] « OTRS Portal -Using the web installer ». [En ligne]. Disponible sur : http://doc.otrs.com/doc/manual/admin/3.3/en/html/web-installer.html

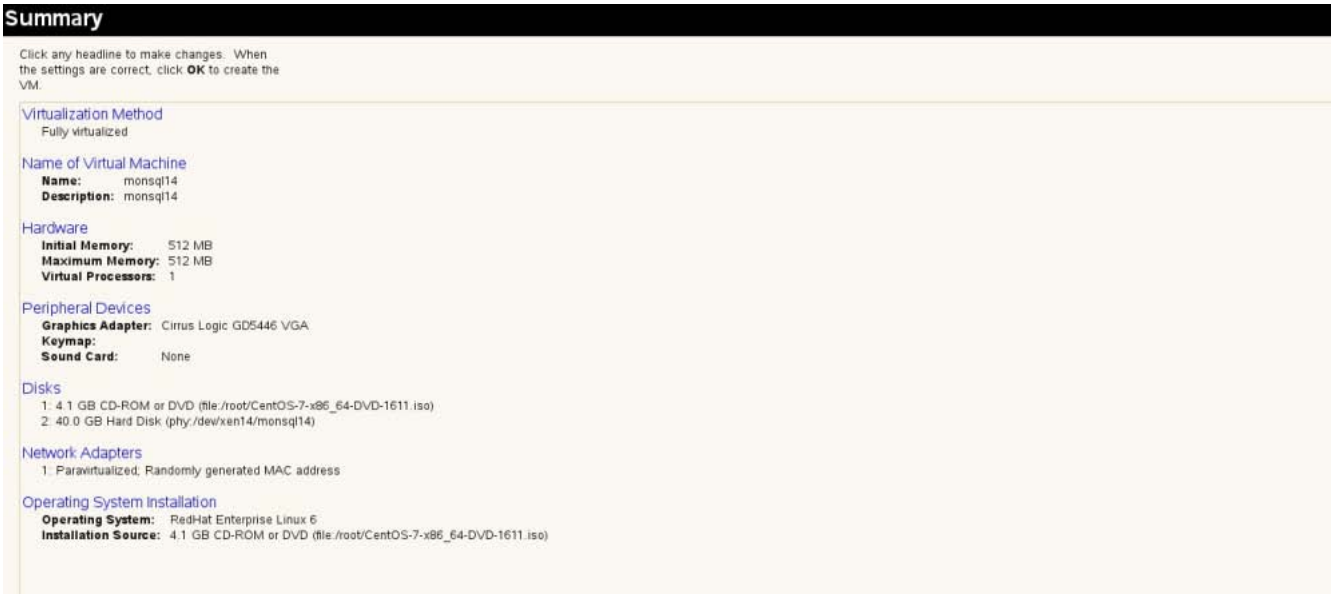
[24] « Installing on RPM-based Linux (CentOS, Fedora, OpenSuse, RedHat) ». [En ligne]. Disponible sur : http://docs.grafana.org/installation/rpm/

ANNEXES

Les étapes de l'installation du système d'exploitation CentOs7 sont décrites dans les figures ci-dessous.



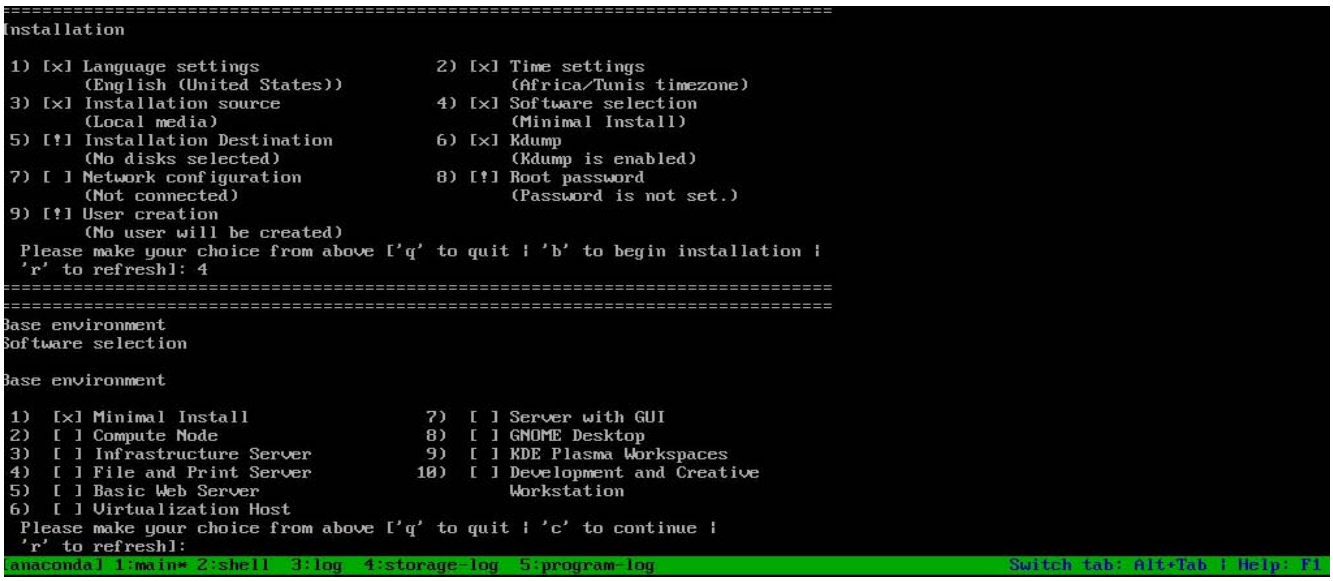
Début de l'installation de CentOs 7



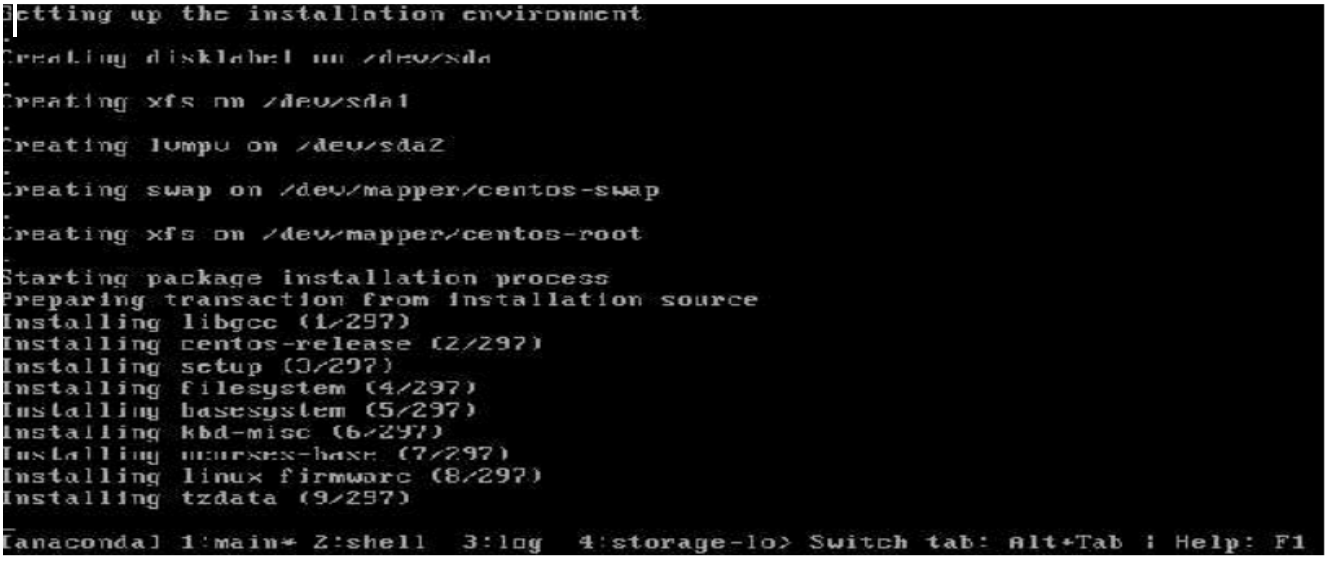
Lors de installation nous devons appuyer sur 'échap' pour pouvoir choisir le mode texte, il faut mentionner la chaine « linux texte » devant le paramètre boot (figure suivante).

La page suivante présente un menu de configuration du système à installer (langage, configuration réseau, Timezone....).

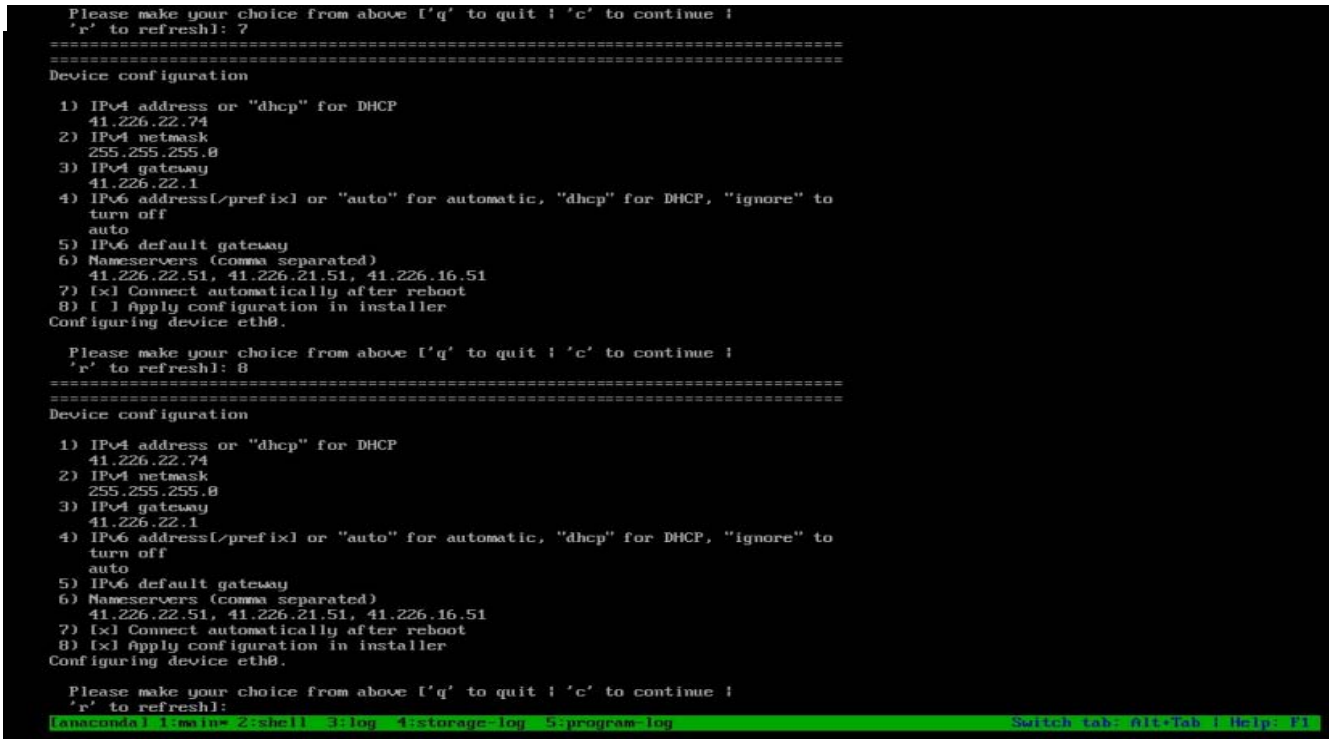
Menu de configuration



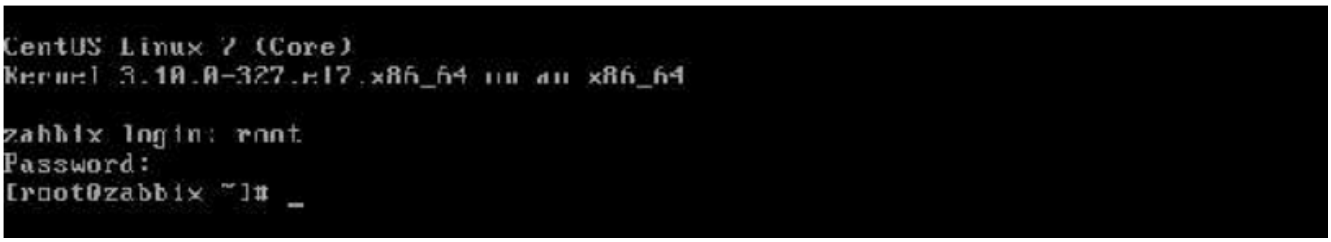
A la fin de la configuration, nous devons appuyer sur b pour lancer l'installation.



Fin de l'installation



1er démarrage



La première opération à effectuer après le premier démarrage est la mise à jour du système d'exploitation avec la commande **yum update**.

Installations des outils de gestion réseauPour faciliter la configuration et la gestion des interfaces réseau, nous avons installé les packages suivants : yum provides ifconfig
yum install net-tools
Configuration du NTP

Afin de synchroniser notre serveur avec un serveur de temps, voici la démarche à entreprendre :

yum install ntp
 # vim /etc/ntp.conf // configuration du service NTP

restrict 192.168.1.0 netmask 255.255.255.0 nomodify notrap
ntpdate -q 0.ro.pool.ntp.org 1.ro.pool.ntp.org
systemctl start ntpd**//démarrage de service**
systemctl enable ntpd// **activation de service au démarrage de système**
#systemctl status ntpd//**status du service ntpd**

Désactivation du firewall par défaut et configuration du firewall privé pour la plateforme Topnet.

Pour commencer il faut arrêter le firewall de la machine pour pouvoir travailler sans restriction de sécurité.

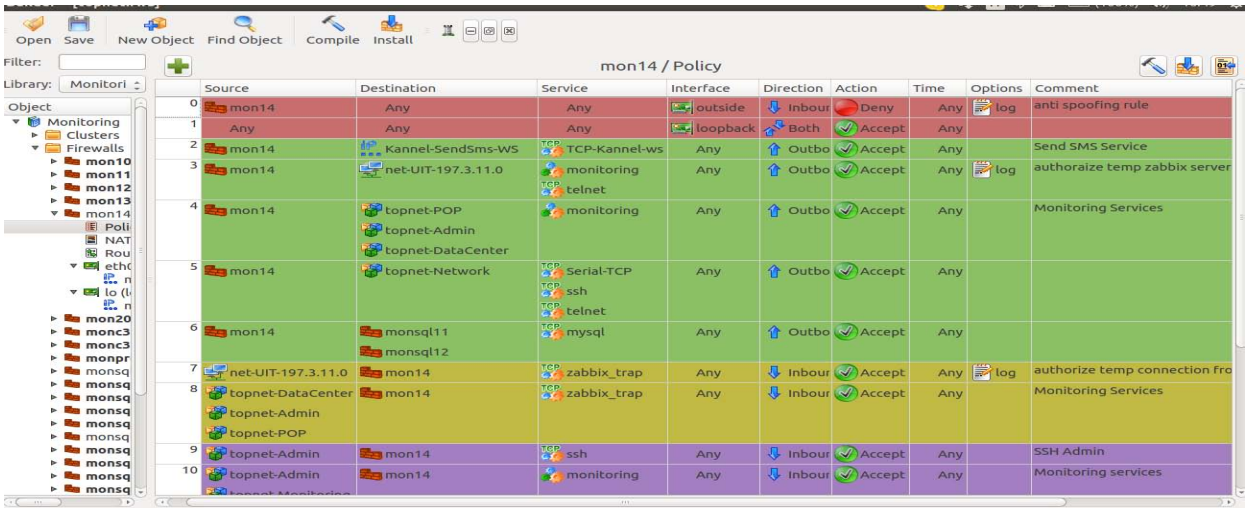
systemctl stop firewalld // arrêter le firewall

systemctl disable firewalld // désactiver au démarrage système

vi /etc/init.d/firewall //
chmod +x /etc/init.d/firewall //
mkdir /etc/firewall //
/etc/init.d/firewall status //

/etc/init.d/firewall stop //
/etc/init.d/firewall status//

Ensuite, il faut ajouter les règles adéquates pour l'autorisation de certaines services et destinations au niveau du fwbuilder.

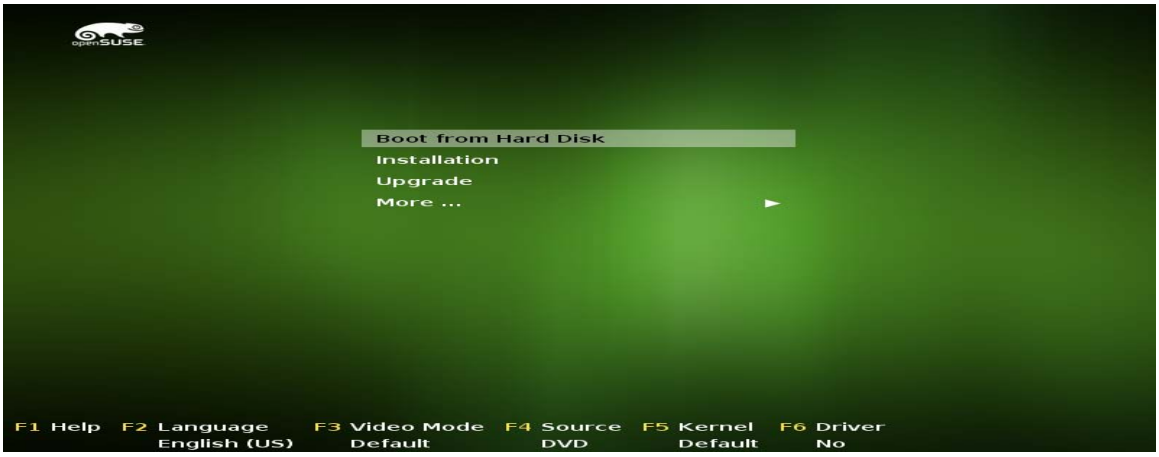


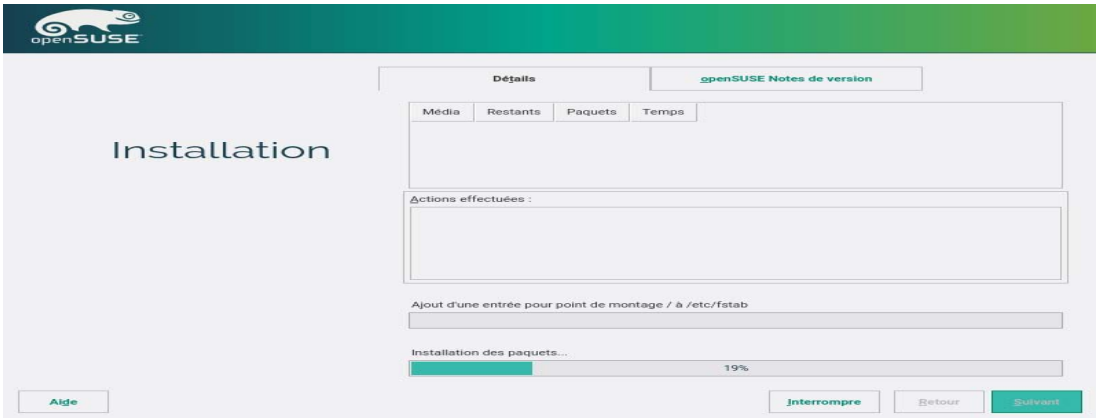
Ci-dessous est une capture d'écran du terminal au niveau du serveur ZabbixWeb

```
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
   Active: inactive (dead)
     Docs: man:firewalld(1)
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]# systemctl desable firewalld
Unknown operation 'desable'.
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]# ls -lash /etc/init.d/firewall
4.0K -rw-r--r--. 1 root root 2.3K 10:42 6 ابر l /etc/init.d/firewall
[root@monweb14 ~]# vi /etc/init.d/firewall
[root@monweb14 ~]# ls /etc/firewall
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]# vi /etc/init.d/firewall
[root@monweb14 ~]# vi /etc/init.d/firewall
[root@monweb14 ~]# vi /etc/init.d/firewall
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]#
[root@monweb14 ~]# rcfirewall start
-bash: /usr/sbin/rcfirewall: Permission denied
[root@monweb14 ~]# chmod +x /etc/init.d/firewall
[root@monweb14 ~]# rcfirewall start
Activating firewall script generated Thu Apr  6 10:43:44 2017 by hanene
Verifying interfaces: eth0 lo
Running prolog script
Rule 0 (eth0)
Rule 1 (lo)
Rule 2 (global)
Rule 3 (global)
Rule 4 (global)
Rule 5 (global)
Rule 6 (global)
Rule 7 (global)
Rule 8 (global)
Rule 9 (global)
Rule 10 (global)
Rule 11 (global)
Rule 12 (global)
Running epilog script
[root@monweb14 ~]# logout
```

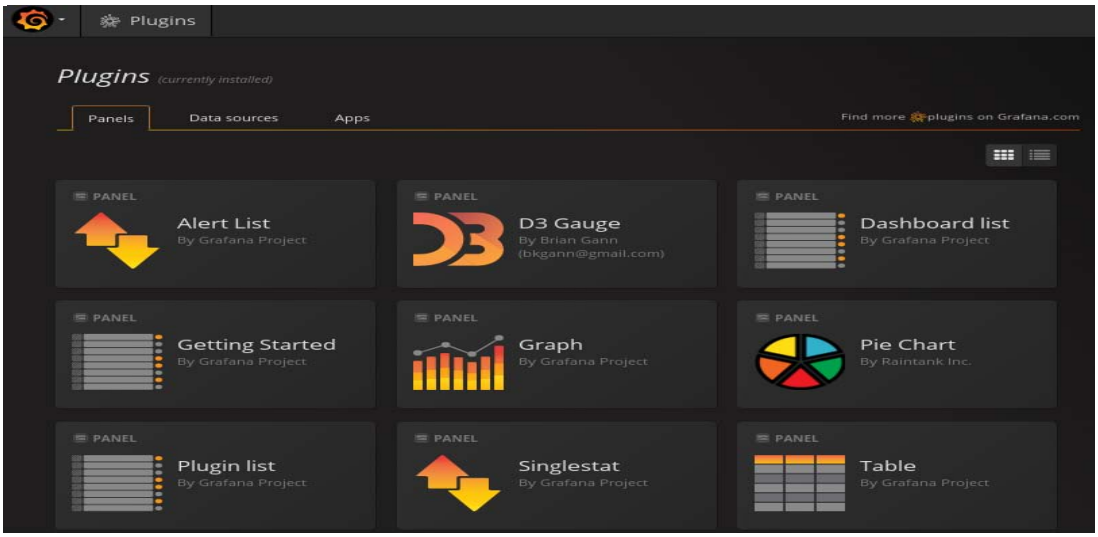
Les étapes de l'installation du système d'exploitation Opensuse est décrite dans les figures ci-dessous :



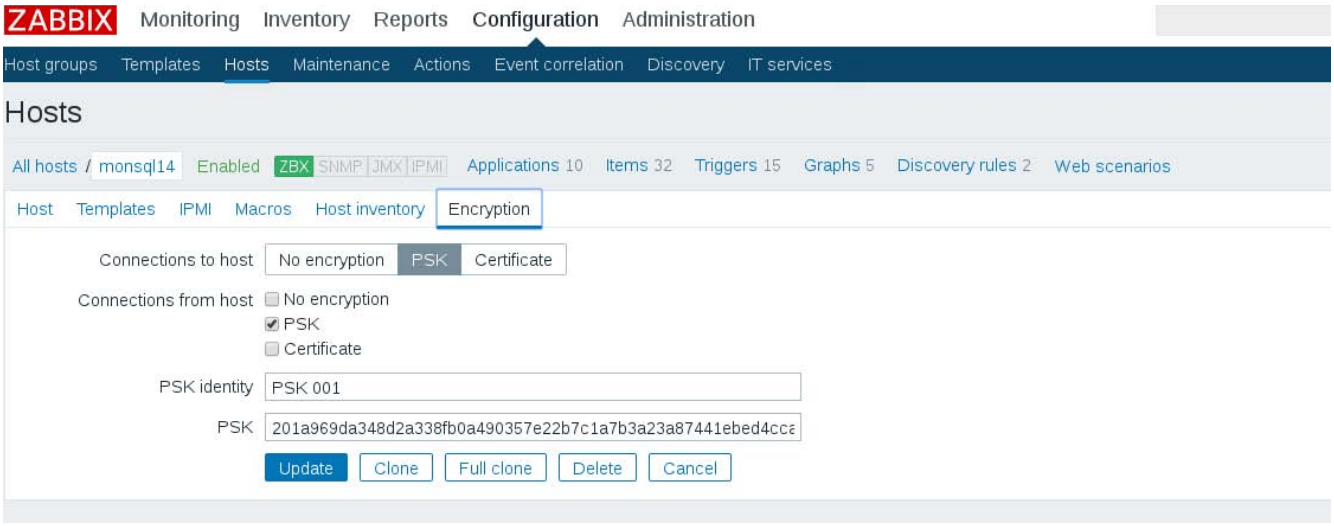


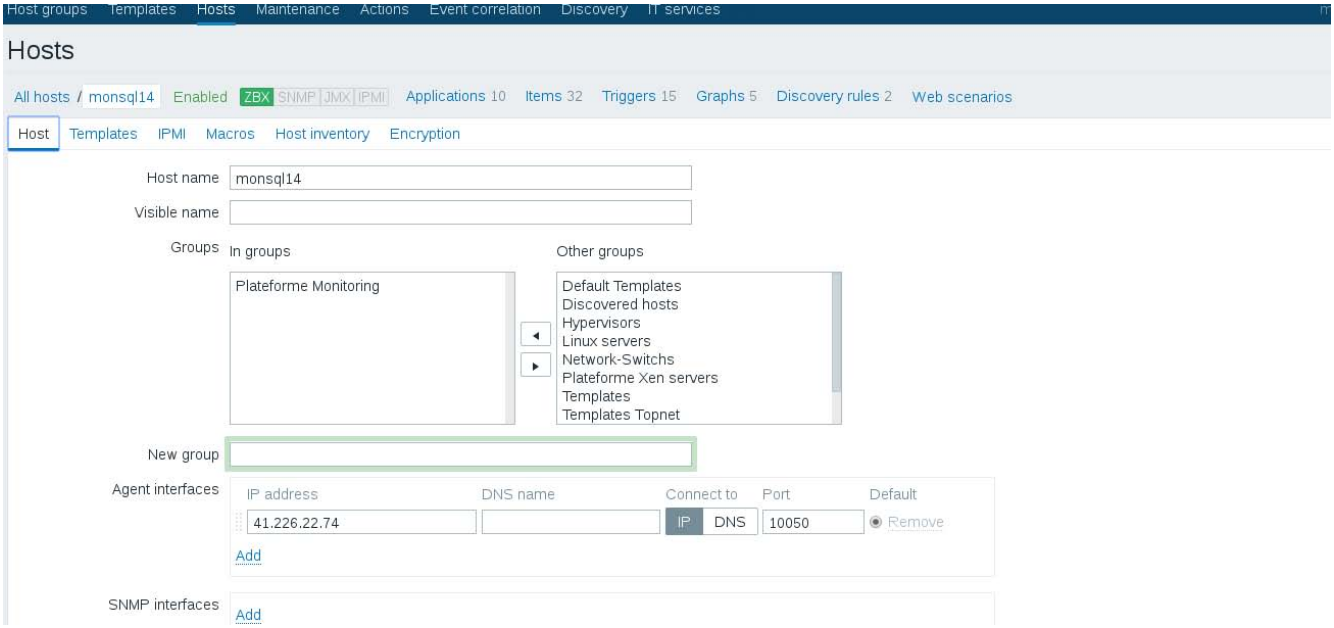
Exemple d'ajout des pannels à Grafana

2017-05-08 09:09:50 grafana-cli plugins install grafana-piechart-panel
2017-05-08 09:17:23 sudo service grafana-server restart
2017-05-08 09:49:11 grafana-cli plugins install briangann-gauge-panel
2017-05-08 09:49:18 sudo service grafana-server restart



Ajout d'un hôte avec l'utilisation de l'encryption pour le Zabbix-agent


ZABBIX
Monitoring
Inventory
Reports
Configuration
Administration
Host groups
Templates
Hosts
Maintenance
Actions
Event correlation
Discovery
IT services
Hosts
All hosts / monsql14
Enabled
ZBX SNMP JMX IPMI
Applications 10
Items 32
Triggers 15
Graphs 5
Discovery rules 2
Web scenarios
Host
Templates
IPMI
Macros
Host inventory
Encryption
Connections to host
No encryption
PSK
Certificate
Connections from host
No encryption
PSK
Certificate
PSK identity
PSK 001
PSK
201a969da348d2a338fb0a490357e22b7c1a7b3a23a87441ebed4cca
Update
Clone
Full clone
Delete
Cancel


Host groups
Templates
Hosts
Maintenance
Actions
Event correlation
Discovery
IT services
Hosts
All hosts / monsql14
Enabled
ZBX SNMP JMX IPMI
Applications 10
Items 32
Triggers 15
Graphs 5
Discovery rules 2
Web scenarios
Host
Templates
IPMI
Macros
Host inventory
Encryption
Host name
monsql14
Visible name
Groups
In groups
Other groups
Plateforme Monitoring
Default Templates
Discovered hosts
Hypervisors
Linux servers
Network-Switchs
Plateforme Xen servers
Templates
Templates Topnet
New group
Agent interfaces
IP address
DNS name
Connect to
Port
Default
41.226.22.74
IP
DNS
10050
Remove
Add
SNMP interfaces
Add