SECS1026

Projet de Stage CCNB

OpenNMS + Grafana

Intervenants : Maryssa LeBlanc, Daren Thibodeau, Agnes Sanama et Mathis Cayouette

Chargés de projet : Anthony Roy, Meriem Oultache et Christian Kalla

Client : Denis Landry

Date : Juin le 6, 2025

Table des matières

Introduction

Dans un contexte où les besoins en supervision des infrastructures TI deviennent de plus en plus critiques, la mise en place d'une solution de monitoring robuste, évolutive et ouverte est essentielle pour assurer la disponibilité, la performance et la sécurité des services numériques d'une organisation. Ce projet d'intégration s'inscrit dans une démarche de modernisation de la supervision réseau, visant à remplacer la solution propriétaire PRTG par une alternative Open Source complète et performante.

L'architecture proposée repose sur le déploiement de la plateforme OpenNMS, intégrée à Grafana pour la visualisation et à Helm pour faciliter l'orchestration et la gestion dans un environnement Kubernetes. Ce choix permet de répondre efficacement aux exigences de supervision de plus de 1000 capteurs, en assurant une grande flexibilité, une gestion fine des alertes, ainsi qu'une intégration fluide avec une page d'état WordPress dédiée.

Au cœur de ce projet, l'objectif est de garantir la continuité des services et le respect du triptyque confidentialité, intégrité et disponibilité (CIA) des données, à travers une supervision proactive, une gestion structurée des accès et une interface utilisateur intuitive. Nous démontrerons ainsi comment une solution bien conçue et bien intégrée permet de répondre aux défis de la supervision moderne tout en respectant les meilleures pratiques en cybersécurité et en gouvernance technique.

Objectifs specifique

- Identifier et déployer un outil de monitoring open source adapté aux besoins de l'organisation.
- Assurer la supervision de plus de 1000 capteurs répartis sur différents services et équipements.
- Garantir la disponibilité, la fiabilité et la performance du système de supervision.
- Permettre l'intégration via API avec une page d'état des services TIC.
- Mettre en place un système d'alerte multi-niveaux en fonction des types d'actifs ou de services.

Aperçu du projet

Ce projet vise à concevoir et à déployer une solution de supervision réseau complète, fiable et évolutive, reposant sur l'intégration d'OpenNMS, Grafana et Helm dans un environnement virtualisé. Cette infrastructure est conçue pour assurer la surveillance de plus de 1000 capteurs répartis sur différents services et équipements informatiques, dans

le but de garantir une visibilité complète sur l'état de santé des systèmes, leur disponibilité, ainsi que la réactivité face aux incidents.

L'architecture du projet repose sur un environnement segmenté hébergé dans Proxmox, incluant :

- Des hôtes Linux et Windows simulant différents types de services critiques;
- Des conteneurs configurés pour des outils complémentaires (ex. Passbolt, Authentik);
- Une instance Grafana pour la visualisation centralisée des indicateurs clés.

La plateforme OpenNMS est déployée à l'aide de Helm dans un environnement Kubernetes afin d'automatiser et de faciliter la gestion des services. L'intégration avec Grafana permet d'afficher des tableaux de bord dynamiques pour une lecture claire et en temps réel des performances réseau et système. Une API REST est également configurée afin de publier une page d'état des services TIC sur WordPress, accessible au public ou restreinte selon les besoins.

Afin de garantir l'efficacité de cette solution, des scénarios de tests de charge, de disponibilité et de fiabilité seront exécutés. Ce projet a été initié en réponse à la volonté de remplacer le logiciel propriétaire PRTG par une solution Open Source, mieux adaptée aux enjeux budgétaires, technologiques et de souveraineté numérique. Il répond également à un besoin croissant de centralisation de la supervision, de gestion proactive des alertes et d'extensibilité des capacités de surveillance.

Enfin, ce projet permet à l'organisation de renforcer sa résilience opérationnelle tout en assurant la conformité avec les bonnes pratiques en cybersécurité. Il contribue à une meilleure anticipation des incidents, à une réaction plus rapide en cas de panne, et à une visibilité améliorée pour les équipes techniques grâce à une supervision modernisée, intuitive et performante.

OpenNMS + Grafana/Helm

Supervision réseau : OpenNMS, Helm et Grafana

La gestion des infrastructures réseau complexes nécessite des outils robustes et flexibles capables de détecter les pannes, d'assurer la collecte de performances et de fournir une visualisation avancée des données. OpenNMS, Helm, et Grafana sont trois solutions complémentaires pour répondre à ces besoins.

OpenNMS : Une platforme open source de supervision réseau

OpenNMS est une plateforme open source conçue pour surveiller, visualiser et gérer des infrastructures informatiques à grande échelle. Capable de superviser plus de 50 000 dispositifs distincts, elle propose des fonctionnalités avancées comme la gestion des événements, le provisionnement automatisé et la collecte de données.

Le système repose sur une architecture performante qui centralise la gestion des événements (traps SNMP, syslogs, XML personnalisés, etc.) pour détecter les pannes et réduire les alarmes redondantes. OpenNMS offre également une supervision des services réseau, allant de simples tests ICMP à des analyses complexes de disponibilité, tout en collectant des données de performance via des protocoles variés comme SNMP, HTTP, JMX ou WMI.

Les données collectées sont visualisées sous forme de tableaux de bord, de cartes thermiques et de rapports de tendances, permettant une analyse détaillée de l'état du réseau. De plus, OpenNMS s'intègre parfaitement avec Grafana grâce au plugin Helm, qui permet une visualisation avancée des métriques et une corrélation efficace des données.

Helm : Simplification du déploiement Kubernetes

Helm est un gestionnaire de packages pour Kubernetes qui facilite le déploiement et la gestion des applications en utilisant des "charts". Ces derniers contiennent tous les fichiers nécessaires au déploiement, tels que des modèles YAML et des paramètres personnalisables. Helm permet de versionner chaque déploiement, ce qui simplifie les mises à jour et les retours en arrière.

Avec Helm, la configuration des ressources devient réutilisable, ce qui le rend idéal pour des environnements nécessitant des déploiements rapides et fiables. Dans le cadre de l'intégration entre OpenNMS et Grafana, Helm est utilisé pour gérer le plugin permettant d'afficher les métriques réseau collectées par OpenNMS directement dans Grafana.

Grafana : Plateforme de visualisation et d'observabilité

Grafana est une solution open source dédiée à la visualisation des données et à l'observabilité. Elle permet de créer des tableaux de bord dynamiques connectés à une multitude de sources de données comme OpenNMS, Prometheus, InfluxDB, et bien d'autres.

Grâce à son interface personnalisable, Grafana offre des graphiques interactifs et des systèmes d'alerte avancés qui aident à surveiller les infrastructures IT, les applications et les performances réseau en temps réel. Les métriques peuvent être analysées en

profondeur pour détecter les problèmes, suivre les tendances et corréler les événements avec des journaux système.

L'intégration entre OpenNMS et Grafana via Helm permet de créer des tableaux de bord riches et dynamiques, où les données de monitoring réseau sont affichées en temps réel. Les administrateurs peuvent ainsi visualiser des informations clés, comme la latence, le taux de disponibilité et les alertes critiques, tout en partageant facilement les résultats avec leur équipe.

Installation et intégration : Pour configurer ces outils ensemble

1. Installer Grafana : Disponible sous Linux, il suffit de configurer le dépôt officiel, d'installer le paquet et de démarrer le service.
2. Installer le plugin Helm dans Grafana : Depuis l'interface de Grafana, ajouter OpenNMS comme source de données.
3. Configurer OpenNMS et Helm : Déployer les configurations nécessaires pour visualiser les métriques dans Grafana.

Remplacement de PRTG par OpenNMS Horizon + Helm + Grafana

L'objectif est de remplacer PRTG par une solution Open Source capable de superviser plus de 1000 capteurs, intégrée à une page d'état WordPress et dotée d'un système d'alertes multi-niveaux.

Solutions analysées :

Critères	OpenNMS + Grafana	Zabbix + Grafana	Prometheus + Grafana
Licence	Open Source (GPL)	Open Source (GPL)	Open Source (Apache 2.0)
Supervision	SNMP, ICMP, HTTP, JMX, WMI	SNMP, ICMP, HTTP, agents Zabbix	Exporters, Pull model (HTTP)
Extensibilité / Plugins	Modules OSGi, scripts, REST	Scripts personnalisés, API REST	Exporters, PromQL
Interface Web	Simple, améliorable par Grafana	Complet et intégré	Basique sans Grafana
Gestion des alertes	Très complète, escalade	Très riche, escalade et délais	Basique sans Alertmanager
API REST	Oui (REST et GraphQL)	Oui	Oui (Prometheus + Alertmanager)
Courbe d'apprentissage	Moyenne à élevée	Moyenne	Élevée (plus technique)
Échelle >1000 capteurs	✔	✔	✔

Après l'analyse de plusieurs alternatives, le choix de la combinaison OpenNMS + Grafana + Helm s'est imposé comme la meilleure réponse aux besoins du projet en matière de supervision.

- OpenNMS Horizon est une plateforme éprouvée, robuste et extensible, compatible avec les protocoles standards tels que SNMP, ICMP et HTTP, ce qui garantit une couverture complète et fiable des équipements. Elle offre également des fonctionnalités avancées de découverte automatique, génération d'alertes, et collecte de performances, le tout sans nécessiter l'ajout de composants supplémentaires comme c'est le cas avec Prometheus.

- Grafana enrichit la solution en apportant des tableaux de bord modernes, interactifs et personnalisables, facilitant la visualisation des indicateurs clés de performance (KPI) pour les administrateurs et utilisateurs.

- Helm, utilisé dans un environnement Kubernetes, permet un déploiement rapide, modulaire et réplicable de l'infrastructure, tout en assurant une gestion automatisée et évolutive des composants.

Cette combinaison de technologies se distingue par son adaptabilité aux environnements cloud-native, sa modularité, et son potentiel d'évolution, contrairement à des solutions comme Zabbix, plus rigide, ou Prometheus, qui requiert l'intégration de plusieurs outils tiers pour atteindre un niveau similaire de fonctionnalité.

Par ailleurs, l'API REST d'OpenNMS permet une intégration fluide avec WordPress, offrant la possibilité d'afficher un état des services TIC de façon publique ou restreinte, selon les besoins de l'organisation.
Enfin, la solution est hautement extensible : elle supporte l'ajout de modules personnalisés, scripts, connecteurs, et s'adapte facilement à l'évolution des exigences métier.

Tableau comparatif détaillé : PRTG vs OpenNMS

Critère	PRTG Network Monitor	OpenNMS	Justification / Explication
Licence	Propriétaire (freemium jusqu'à 100 capteurs)	Open source (GPL v3)	OpenNMS est gratuit et open source, ce qui permet une personnalisation avancée et aucune limitation de capteurs, contrairement à PRTG qui devient payant après 100 capteurs.
Coût	Payant (selon le nombre de capteurs)	Gratuit	Pour un grand nombre de capteurs (1000+), OpenNMS est plus économique, tandis que PRTG peut devenir coûteux.
Scalabilité	Moyenne	Élevée (architecture	OpenNMS peut s'adapter à de très grandes infrastructures grâce à son

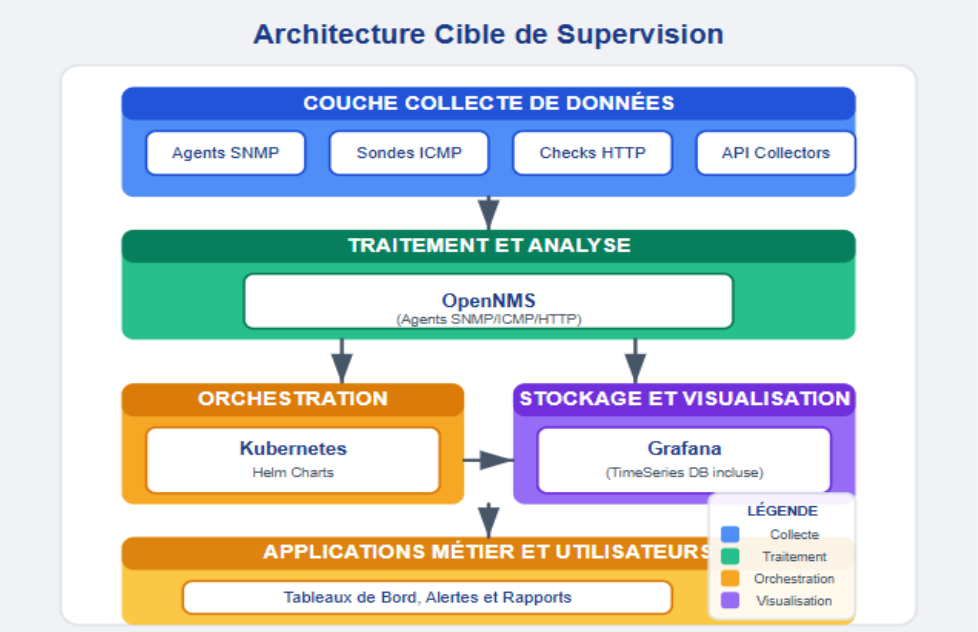
		distribuée possible)	architecture distribuée (Minions, Kafka, etc.), ce qui est plus complexe à faire avec PRTG.
Installation	Simple, rapide (Windows uniquement)	Plus complexe (Linux, Java, PostgreSQL requis)	PRTG est clé en main sur Windows, tandis que OpenNMS requiert des compétences système (Linux, Java, PostgreSQL), mais offre plus de contrôle.
Support de plateforme	Windows uniquement	Linux (Debian, Ubuntu, CentOS)	OpenNMS est nativement compatible Linux, ce qui s'aligne bien avec des environnements serveurs modernes, tandis que PRTG est limité à Windows.
Interface utilisateur	Intuitive, moderne, glisser-déposer	Interface Web technique mais personnalisable	L'interface PRTG est plus conviviale pour les débutants. OpenNMS offre une interface plus technique mais modulable via Grafana et Helm.
Protocoles supportés	SNMP, WMI, HTTP, Ping, NetFlow	SNMP, HTTP, ICMP, JMX, XML, NetFlow, Syslog, REST	Les deux supportent SNMP, mais OpenNMS couvre plus de protocoles (ex : REST API, Syslog, JMX) et permet une meilleure intégration avec des outils externes.
Plugins & Extensibilité	Limitée à l'écosystème Paessler	Très élevée (API REST, modules Java, Grafana, etc.)	OpenNMS est hautement extensible (Helm, Grafana, Minions, etc.) grâce à son architecture modulaire. PRTG est plus fermé à l'extérieur de ses plugins officiels.
Dashboards personnalisés	Oui (internes à PRTG)	Oui (via Grafana + Helm)	OpenNMS + Grafana permet des dashboards très puissants et dynamiques, tandis que ceux de PRTG sont plus limités mais faciles à utiliser.
Alertes & notifications	Avancées, configurables, faciles à utiliser	Très avancées, via règles XML, scripts, API	OpenNMS permet une gestion fine et scriptable des alertes, adaptée aux environnements complexes, mais demande plus de configuration.
Auto-découverte réseau	Oui (très intuitive)	Oui (via règles de provisioning)	OpenNMS a une auto-découverte intelligente, mais plus technique à configurer. PRTG est plus rapide à mettre en œuvre pour une découverte initiale.
Supervision de flux réseau	Oui (via NetFlow, sFlow, jFlow)	Oui (OpenNMS Flow avec Collectd / Telemetryd)	Les deux offrent la collecte de flux, mais OpenNMS Flow nécessite une configuration avancée (via Kafka ou gRPC), ce qui permet aussi des corrélations poussées.

Supervision de services cloud	Support limité	Support avancé (Cloud monitoring, API)	OpenNMS est plus adapté au monitoring des environnements hybrides (cloud, conteneurs, microservices).
Supervision IoT / SNMPv3	Partiel (peu d'options avancées SNMPv3)	Support complet SNMPv1/v2c/v3 + Trap Receiver	OpenNMS est plus adapté à des environnements industriels/IoT, notamment avec SNMPv3 sécurisé.
Export de données / Historique	Export limité (CSV/Excel via interface)	Export complet via PostgreSQL / API	OpenNMS donne un accès direct à la base PostgreSQL, facilitant les analyses avancées, exports, ou intégrations BI.
Communauté & Support	Support commercial (Paessler), forum	Communauté active + entreprise The OpenNMS Group	OpenNMS a une grande communauté open source, avec un support pro disponible via The OpenNMS Group. PRTG est soutenu uniquement par son éditeur.

Justification pour un projet de migration

Aspect	PRTG	OpenNMS (choix recommandé)
Coût pour 1000+ capteurs	Élevé (licence nécessaire)	Gratuit
Personnalisation	Limitée	Très élevée (tableaux Grafana, scripts, API REST, configuration fine)
Écosystème open source	Non	Oui (Grafana, PostgreSQL, Prometheus, InfluxDB, etc.)
Compétences requises	Faibles (interface clé-en-main)	Élevées (Linux, réseau, SNMP, PostgreSQL, YAML/XML)
Interopérabilité ITSM	Faible	Haute (Intégration avec CMDB, Jira, outils DevOps via API)
Justification du choix	Solution rapide à déployer	Idéal pour des besoins d'échelle, de personnalisation, d'intégration et de maîtrise complète de la supervision réseau.

Architecture cible



Composition des couches fonctionnelles :

Couche de collecte de données

- Agents SNMP : Pour la supervision des équipements réseau
- Sondes ICMP : Pour les tests de connectivité et de latence
- Checks HTTP : Pour la supervision des applications web
- API Collectors : Pour l'intégration avec d'autres systèmes

Cette couche forme toujours la base de votre système en récupérant les métriques depuis vos différents équipements.

Couche de traitement et analyse

- OpenNMS : Composant central qui :
  - Coordonne les agents SNMP/ICMP/HTTP
  - Gère les événements et alertes
  - Effectue le traitement initial des données
  - Transmet les données vers la couche de stockage

Couche d'orchestration

- Kubernetes (K8s) : Plateforme d'orchestration des conteneurs
- Helm Charts : Système de gestion des packages pour Kubernetes
  - Facilite l'installation et la mise à jour des composants
  - Gère les configurations et dépendances

Couche de stockage et visualisation

- Grafana : Solution intégrée qui :
    - Stocke les données temporelles (remplace la base TimeSeries indépendante)
    - Fournit des tableaux de bord personnalisables
    - Propose des systèmes d'alertes visuelles
    - Permet la visualisation des métriques collectées

Couche applications métier et utilisateurs

- Représente les utilisateurs et applications qui consultent les tableaux de bord
- Point d'accès unifié pour la supervision de l'infrastructure

Principaux flux de données dans cette architecture

1. Les agents de collecte recueillent les données depuis les équipements
2. OpenNMS centralise et traite ces données
3. Helm/Kubernetes gère le déploiement et la configuration de l'infrastructure
4. Grafana stocke les données temporelles et fournit les visualisations
5. Les utilisateurs et applications accèdent aux tableaux de bord pour consulter l'état du système

Avantages de cette architecture révisée

1. Simplification: Élimination de la base de données temporelle séparée au profit de la solution intégrée de Grafana
2. Cohérence: Meilleure intégration entre Helm et Grafana pour le stockage des données
3. Flexibilité : Possibilité d'étendre ou de modifier chaque couche indépendamment
4. Évolutivité : Architecture conteneurisée facilitant le scaling horizontal
5. Maintenance : Déploiement et mises à jour simplifiés via Helm Charts

Points à considérer pour l'implémentation

1. Dimensionnement de Grafana : Assurez-vous que votre déploiement Grafana est correctement dimensionné pour gérer à la fois la visualisation et le stockage des séries temporelles
2. Sauvegardes : Établissez une stratégie de sauvegarde pour les données de Grafana
3. Intégration OpenNMS-Grafana : Configurez correctement les connecteurs entre OpenNMS et Grafana
4. Sécurité : Mettez en place les contrôles d'accès appropriés pour chaque composant

Prérequis techniques

Composant	Exigences
OS	Ubuntu Server 22.04 LTS / Debian / Proxmox
Matériel	4 CPU, 8-16 Go RAM, 100+ Go SSD
Réseau	Accès SNMP, HTTP/S, ICMP, ports Grafana, API
Alerting	SMTP, webhook, Slack, SMS via passerelle
API	REST (pour page WordPress)

Inventaire des capteurs

Dans l'environnement actuel, la majorité des capteurs utilisés dans PRTG sont de type ping, permettant de vérifier la disponibilité des équipements réseau. Des capteurs SNMP complètent cette surveillance en analysant les ports des switches et pare-feux, avec des alertes configurées en cas de dépassement de seuils. D'autres capteurs suivent l'état de santé global des équipements (mémoire, processeur, stabilité). Une attention particulière est portée à la page « État des services TIC », qui utilise des capteurs ping et HTTP pour valider automatiquement, via les API de PRTG, la disponibilité des services (comme un retour HTTP 200 pour les sites web). Les alertes sont acheminées selon les services concernés : réseau, caméras, Alertus, etc.

Dans le cadre de la migration vers une solution de supervision open source comme OpenNMS, il est essentiel de réaliser un inventaire structuré des capteurs. Cela permet de s'assurer que tous les services critiques seront bien couverts après la migration. En classant et regroupant les capteurs, on facilite la planification, on élimine les doublons et on priorise les éléments essentiels. Ce processus garantit aussi une meilleure visibilité de l'environnement supervisé et pose les bases d'une documentation claire et à jour.

La page "État des services TIC" semble afficher les services supervisés en temps réel, y compris leur état fonctionnel, intermittent ou non-fonctionnel. Voici les éléments généralement supervisés selon votre infrastructure et les informations trouvées :

1. Accessibilité réseau : Vérifié via des capteurs Ping ou HTTP (par exemple, pour garantir un code HTTP 200).
2. Ressources système : CPU, mémoire, stockage des serveurs, switches et firewalls.
3. Applications critiques : Par exemple, TOPdesk ou le site web du CCNB.
4. Infrastructure Wi-Fi : Points d'accès et leur connectivité.

5. Intégration API : Interaction entre PRTG et des systèmes externes, comme WordPress.

Ces services sont rapportés en temps réel sur la page pour que la communauté puisse les consulter avant de solliciter le support informatique.

https://etatdesservicestic.ccnb.ca/

Inventaire des capteurs PRTG en production

→ Tableau d'inventaire des capteurs (sensors) :

Type de capteur	Catégorie	Description	Critères d'alerte	Groupes d'avis
Ping	Disponibilité réseau	Détermine si l'appareil est sur le réseau et accessible.	Appareil non accessible.	Analystes réseau, Équipe technique
SNMP	Santé des équipements	Surveillance de la mémoire, du CPU, et autres paramètres de santé des switches et firewalls.	Utilisation mémoire/CPU au-dessus d'un seuil défini.	Analystes réseau, Équipe technique
HTTP	État des services TIC	Valide l'état des services TIC (ping + réponse HTTP avec code 200).	Non-réponse ou code HTTP différent de 200.	Équipe technique
CPU	Santé des équipements	Surveillance de l'utilisation du CPU des équipements.	Utilisation CPU au-dessus d'un seuil défini.	Analystes réseau, Équipe technique
RAM	Santé des équipements	Surveillance de l'utilisation de la mémoire vive (RAM) des équipements.	Utilisation mémoire vive au-dessus d'un seuil défini.	Analystes réseau, Équipe technique
Disque	Santé des équipements	Surveillance de l'utilisation	Utilisation disque au-	Analystes réseau,

		du stockage disque.	dessus d'un seuil défini.	Équipe technique
Bande passante	Utilisation des ports	Monitorer l'utilisation des ports sur les switches et firewalls.	Niveau de trafic au-dessus d'un seuil défini.	Analystes réseau, Équipe technique
Wi-Fi	Réseaux Wi-Fi	Surveillance des points d'accès (par ex. 50 sur 250 AP hors ligne entraîne une alerte pour perturbation).	Déconnexion d'un pourcentage défini des points d'accès.	Analystes réseau
API	Intégration API	Intégration avec WordPress pour afficher l'état des services (fonctionnel, perturbé ou non fonctionnel).	État fonctionnel avec perturbations ou non-fonctionnel détecté.	Équipe technique,

Détails des capteurs

- Ping : Ces capteurs sont majoritairement utilisés (90%) pour déterminer si les appareils sont accessibles sur le réseau.
- SNMP : Utilisé pour surveiller les ressources des équipements tels que la mémoire et le CPU.
- HTTP : Utilisé pour valider les réponses des services Web (exemple : code HTTP 200 attendu).
- CPU : Capteurs spécifiques pour monitorer l'utilisation du processeur.
- RAM : Capteurs pour surveiller l'utilisation de la mémoire vive.
- Disque : Capteurs pour surveiller l'espace de stockage et son utilisation.
- Bande passante : Capteurs pour monitorer le trafic sur les ports des switches et firewalls.
- Wi-Fi : Surveillance des points d'accès et de leur disponibilité.
- API : Utilisé pour l'intégration avec des systèmes externes comme WordPress.

Processus d’alerte

Les alertes sont gérées en fonction de la classe des appareils :

- Switches : Alertes envoyées aux analystes réseau.
- Caméras : L'équipe de maintenance est avisée.
- Systèmes locaux : L'équipe technique reçoit les notifications.
- Alertus : Le DASA des campus est informé.

Ce système assure une surveillance efficace et une communication rapide avec les équipes concernées pour minimiser les interruptions de service.

Pour identifier les capteurs nécessitant une recréation manuelle, il est nécessaire de prendre en compte plusieurs facteurs, comme les types de capteurs, leurs configurations spécifiques, et les intégrations système.

Capteur nécessitant (possiblement) une recréation manuelle et leurs raisons :

1. PING - Mauvaise configuration de l'adresse ou des seuils, dépendance à des paramètres externes spécifiques

2. SNMP - Problèmes avec l'intégration des MIB, configuration manuelle des hôtes SNMP pour certains équipements.

3. HTTP - Problèmes de configuration des URL, problèmes de certificats SSL ou de ports non standards.

4. CPU - Capteurs spécifiques aux équipements qui ne migrent pas automatiquement.

5. RAM - Configurations des serveurs ou matériels qui varient d'un modèle à l'autre, nécessitant une réintégration.

6. DISQUE – Besoin de spécifications de partitionnement spécifiques ou de seils personnalisés.

7. BANDE PASSANTE - Configuration des interfaces ou des ports particuliers nécessitant une recréation des capteurs.

8. Wi-Fi - Problèmes de compatibilité avec certains points d'accès, nécessitant une configuration manuelle des capteurs.

9. API - API non documentées ou mises à jour, nécessitant des ajustements manuels pour l'intégration.

Environnement de Test : Windows Server 2019, PFsense et Ubuntu

Dans le cadre de ce projet, la mise en place d'un environnement de test fiable et sécurisé est essentielle pour simuler des scénarios réels et évaluer les performances des outils et configurations déployés. Trois composants principaux ont été intégrés : Windows Server 2019, PFsense et Ubuntu. Chacun joue un rôle spécifique dans la construction d'un laboratoire complet et fonctionnel.

Windows Server 2019

Windows Server 2019 est utilisé comme un serveur polyvalent au sein de l'environnement de test. Son rôle principal comprend :

- Gestion des services réseau : Windows Server agit comme un serveur DHCP, DNS ou Active Directory pour gérer les ressources réseau.
- Test des politiques de sécurité : Les stratégies de groupe (GPO) et d'autres configurations de sécurité peuvent être appliquées et évaluées.
- Hébergement de services spécifiques : Par exemple, l'hébergement d'applications critiques ou de bases de données, permettant d'analyser leur comportement dans un environnement simulé.

Dans ce projet, Windows Server 2019 fournit une base pour intégrer et tester les interactions entre différents services et outils, tels qu'OpenNMS et Grafana, dans un environnement typique d'entreprise.

PFsense

PFsense est un pare-feu et un routeur open source basé sur FreeBSD. Il joue un rôle crucial dans la sécurisation et la gestion de l'environnement réseau du laboratoire. Ses fonctions principales incluent :

- Pare-feu et filtrage de trafic : PFsense contrôle rigoureusement le trafic réseau pour garantir une isolation et une sécurité optimales.

- Configuration de VPN : Des connexions VPN peuvent être établies pour un accès sécurisé aux ressources du laboratoire depuis l'extérieur.
- Segmentation du réseau : PFsense permet la création de VLANs pour séparer les différentes parties du réseau, facilitant des tests spécifiques (services, utilisateurs, IoT).
- Monitoring réseau : Grâce à ses outils intégrés, PFsense surveille le trafic réseau pour détecter les anomalies, essentiel pour tester les menaces dans un environnement contrôlé.

Ubuntu

Ubuntu est utilisé comme une plateforme de serveur principale pour déployer des outils critiques et simuler un environnement Linux couramment utilisé dans les entreprises. Son rôle comprend :

- Hébergement des services d'application : Ubuntu est utilisé pour déployer OpenNMS, Grafana et WordPress, offrant une plateforme robuste et évolutive.
- Analyse des vulnérabilités : La configuration d'outils comme DIVA permet de tester des scénarios liés aux menaces de sécurité et de renforcer les configurations.
- Environnement de développement : Ubuntu fournit une base idéale pour exécuter des scripts, configurer des bases de données (PostgreSQL) et gérer les dépendances nécessaires pour le projet.

En utilisant Ubuntu, il est possible de reproduire des environnements de production courants, de tester des configurations réseau et des services, et d'intégrer ces solutions avec d'autres composants de l'infrastructure.

Rôle Global dans le Projet

L'utilisation conjointe de Windows Server 2019, PFsense et Ubuntu dans ce projet permet de :

- Créer un environnement de test complet et réaliste pour déployer et tester les outils (OpenNMS, Grafana, WordPress) dans des conditions variées.
- Simuler des scénarios complexes, tels que des attaques ou des pannes, afin d'évaluer la robustesse et la résilience des configurations réseau et de sécurité.
- Fournir une infrastructure modulaire et adaptable à différents scénarios, notamment l'intégration avec des solutions cloud ou des plateformes IoT.
- Valider les stratégies de sécurité, les performances des services et les politiques de gestion réseau avant tout déploiement en production.

En résumé, cet environnement de test est une base essentielle pour expérimenter, optimiser et valider les solutions de gestion et de sécurité réseau dans le cadre de ce projet.

Installation de la solution

Nous avons procédé à l'installation et à la configuration de Grafana comme interface de visualisation pour la solution de supervision. Grafana a été intégré avec OpenNMS à l'aide du plugin approprié, via Helm, afin d'assurer la génération de tableaux de bord dynamiques et personnalisables.

Installation de Grafana

L'installation de Grafana a été réalisée sur un serveur Ubuntu à l'aide des étapes suivantes

- Ajout du dépôt officiel Grafana.
- Installation du paquet via apt.
- Démarrage et activation du service Grafana
- Ouverture du port 3000 via le pare-feu pour l'accès web à l'interface utilisateur.

Ajout du plugin OpenNMS via Helm

L'intégration d'OpenNMS avec Grafana a été effectuée grâce à l'installation du plugin Helm de OpenNMS, qui permet d'exposer les métriques collectées par OpenNMS sous une forme exploitable dans Grafana :

- Déploiement du plugin via Helm charts.
- Configuration du connecteur dans Grafana pour l'ajout de OpenNMS comme source de données.
- Paramétrage des requêtes de données compatibles avec les panels Grafana.

Configuration des tableaux de bord

Des tableaux de bord personnalisés ont été créés afin de visualiser en temps réel les indicateurs clés de performance (KPI) tels que :

- La disponibilité des services (via les capteurs ICMP, HTTP).
- L'état des équipements (CPU, RAM, stockage).
- Le trafic réseau (via SNMP).
- Les événements critiques (alertes générées par OpenNMS).

Grafana nous a permis d'organiser ces métriques sous forme de graphes, jauges, alertes visuelles, et cartes thermiques, facilitant ainsi l'interprétation rapide de l'état de l'infrastructure.

Avantages de cette configuration

- Interface utilisateur intuitive et responsive.
- Intégration directe avec OpenNMS via API.
- Génération d'alertes conditionnelles directement depuis les dashboards Grafana.
- Possibilité de partager des vues (URL ou export PDF) avec les équipes techniques.

Sécurisation du serveur OpenNMS + Grafana

La sécurisation des systèmes informatiques est cruciale pour protéger les données sensibles, maintenir l'intégrité des services et prévenir les attaques malveillantes. Dans le cadre de l'hébergement de services comme OpenNMS et Grafana sur un serveur Ubuntu 22.04, il est essentiel de suivre des pratiques de sécurité rigoureuses pour garantir que ces plateformes de supervision et de gestion soient protégées contre les vulnérabilités exploitables.

Le premier objectif de cette sécurisation est de renforcer l'accès administratif en modifiant les mots de passe par défaut, en activant des mots de passe forts, et en supprimant les comptes inutilisés. Il est également primordial de créer des comptes utilisateurs avec des privilèges minimaux pour limiter les risques en cas de compromission d'un compte. La gestion des accès par UFW (Uncomplicated Firewall) permet de restreindre les ports utilisés, en autorisant uniquement les accès nécessaires comme SSH (port 22) pour un accès sécurisé ou les interfaces Web d'OpenNMS (port 8980) et Grafana (port 3000).

La sécurisation de Grafana passe par la mise en place d'un mot de passe administrateur robuste, l'activation de l'HTTPS pour chiffrer les communications, et la journalisation des accès pour suivre toute tentative d'intrusion. De même, la surveillance des logs est un aspect fondamental pour détecter les anomalies et prévenir les tentatives d'accès non autorisées.

Le pare-feu UFW doit être configuré pour ne laisser passer que les connexions provenant des IP autorisées, ce qui empêche les attaques par balayage de port ou d'accès non autorisé aux interfaces sensibles. Cette approche réduit l'exposition aux attaques externes tout en garantissant l'accessibilité nécessaire aux administrateurs.

Des recommandations complémentaires sont également proposées pour maintenir la sécurité sur le long terme. Il est essentiel de mettre à jour régulièrement le système et les services afin de corriger rapidement les vulnérabilités découvertes. La désactivation des

services inutiles réduit la surface d'attaque et minimise les risques associés aux services non protégés. La mise en place d'un système de sauvegarde fiable et la vérification périodique des procédures de restauration assurent que les données critiques peuvent être récupérées en cas d'incident.

En parallèle, des outils de surveillance comme AIDE permettent de vérifier l'intégrité des fichiers du système, détectant toute modification non autorisée, tandis que des solutions de journalisation centralisée, telles que rsyslog, permettent de centraliser les logs pour une gestion et analyse facilitées.

L'importance de sécuriser ces systèmes ne peut être sous-estimée, car toute vulnérabilité non corrigée peut être exploitée pour compromettre l'ensemble du réseau ou accéder à des informations sensibles. En appliquant des mesures de sécurité rigoureuses comme celles décrites, on réduit considérablement les risques de compromission, de perte de données ou d'attaque par déni de service, tout en assurant la continuité de service pour les utilisateurs autorisés.

Intégration de Grafana avec le plugin OpenNMS

Grafana est une solution open source largement adoptée pour la visualisation et l'observabilité des données. Couplé à OpenNMS, une plateforme complète de supervision réseau, il permet une surveillance avancée et une corrélation efficace des données critiques de l'infrastructure. Cette annexe détaille les étapes clés pour l'installation et la configuration de cette intégration, en mettant en avant ses avantages et ses applications.

**Pourquoi associer Grafana avec OpenNMS ?

L'intégration de Grafana et OpenNMS combine les points forts des deux solutions. OpenNMS offre une collecte efficace des données, une détection de pannes et une gestion d'événements en temps réel, tandis que Grafana propose une visualisation avancée des métriques, des tableaux de bord flexibles et une interface utilisateur intuitive. Ensemble, ces outils fournissent une vision globale unifiée, une représentation optimisée des données et une corrélation efficace, offrant ainsi une expérience utilisateur enrichie.

Pour assurer une intégration réussie, il est nécessaire de respecter certains prérequis. Cela inclut l'utilisation de versions récentes de systèmes d'exploitation comme Linux, Windows ou macOS. Les ressources matérielles doivent comprendre au moins deux processeurs (CPU) et deux gigaoctets de mémoire vive (RAM).

Installation de grafana et du plugin OpenNMS

L'installation commence par le téléchargement de Grafana à partir de son dépôt officiel. Une fois l'installation effectuée via des commandes telles que "apt install grafana", il convient de démarrer et de vérifier le service à l'aide de "systemctl start grafana-server". Ensuite, le plugin OpenNMS doit être activé dans la section "Plugins" de l'interface de Grafana. Une fois localisé, le plugin peut être activé, et les permissions doivent être configurées pour garantir un accès adapté aux données.

Pour connecter Grafana à OpenNMS, une nouvelle source de données doit être ajoutée dans l'interface de Grafana. L'URL du serveur OpenNMS doit être configurée, en suivant un format typique comme "http://serveur-opennms:8980/opennms". Les paramètres de connexion sont ensuite testés et validés pour s'assurer que tout fonctionne correctement.

Personnalisation et exploitation des tableaux de bord

Les tableaux de bord Grafana offrent une organisation flexible en panneaux adaptés aux besoins de visualisation. Les utilisateurs peuvent tirer parti des variables et des filtres pour personnaliser dynamiquement l'affichage des données. Divers types de visualisations sont disponibles, notamment des graphiques, des tableaux et des étiquettes contextuelles, ce qui permet de présenter les données de manière claire et intuitive. Par exemple, un tableau de bord typique peut afficher des visualisations ICMP pour la disponibilité réseau, des métriques HTTP pour les services applicatifs, et des évolutions des performances pour l'analyse historique et des tendances.

L'intégration entre Grafana et OpenNMS offre une supervision améliorée, combinant des fonctionnalités de monitoring avancé avec une visualisation intuitive. Cette solution permet aux organisations d'optimiser leur réactivité face aux incidents et de renforcer la fiabilité de leurs infrastructures. Pour aller plus loin, la consultation de la documentation officielle de Grafana et OpenNMS est recommandée.

Configuration des hotes et services

La migration des capteurs de PRTG vers OpenNMS nécessite une documentation détaillée des correspondances entre les anciens capteurs PRTG et les nouveaux capteurs OpenNMS. Cette documentation doit être structurée sous forme de tableau et accompagnée d’explications pour chaque type de surveillance, afin de faciliter la migration lors du remplacement de PRTG par OpenNMS. Les principaux types de capteurs concernés incluent la disponibilité réseau (ICMP/Ping), la surveillance des services web (HTTP/HTTPS), la supervision des ressources systèmes (CPU, mémoire, disques), la surveillance du trafic réseau via SNMP, la supervision des ports et services personnalisés,

ainsi que la gestion des alertes SNMP et des équipements spécifiques comme les imprimantes et points d'accès WiFi. Certains capteurs PRTG, tels que ceux basés sur WMI ou la supervision avancée de VMware et Microsoft 365, ne sont pas nativement pris en charge par OpenNMS et nécessitent des scripts ou des intégrations manuelles spécifiques.

La méthodologie de migration se décompose en plusieurs étapes. Il convient d'abord de réaliser un inventaire exhaustif des capteurs PRTG, en exportant la configuration ou en listant les capteurs par hôte. Ensuite, il faut identifier les équivalents dans OpenNMS, en distinguant les capteurs standard (ICMP, SNMP, HTTP) de ceux nécessitant une configuration XML ou un ajout manuel dans les fichiers de configuration d'OpenNMS. Des tests et validations sont ensuite réalisés via l'interface web d'OpenNMS pour vérifier la découverte des services, l'affichage des graphiques de performance et la génération d'alertes. Enfin, la création de tableaux de bord via Grafana permet de visualiser les indicateurs clés et d'ajouter des alertes visuelles dynamiques.

La configuration d'OpenNMS implique plusieurs étapes opérationnelles. Il faut d'abord ajouter les hôtes manuellement ou via la découverte automatique, en s'assurant que chaque hôte apparaisse correctement dans la liste des nœuds et que la supervision soit opérationnelle. Les services supervisés (ping, SNMP, HTTP, etc.) doivent être activés et configurés pour chaque hôte, en vérifiant notamment la connectivité SNMP. L'ajout de nouveaux nœuds peut se faire manuellement ou par détection automatique de plages d'adresses IP, avec une validation finale dans le tableau de bord d'OpenNMS.

Il est également essentiel de documenter la correspondance entre les capteurs PRTG et OpenNMS, en exportant la liste des capteurs existants et en identifiant leurs équivalents dans OpenNMS. Cette documentation doit être maintenue à jour dans un fichier Excel ou un wiki d'entreprise, facilitant ainsi le suivi et la maintenance du système de supervision.

Enfin, la configuration des permissions et des alertes dans OpenNMS doit être rigoureusement appliquée. Les utilisateurs et groupes doivent être créés avec des rôles adaptés (lecture seule, administrateur, etc.), et les permissions doivent être testées pour garantir leur efficacité. La mise en place des alertes permet de surveiller les événements critiques, avec la définition des destinataires et la vérification du bon fonctionnement des notifications en cas d'incident

L'ensemble de ces étapes garantit une migration structurée et efficace de PRTG vers OpenNMS, tout en assurant la continuité et la fiabilité de la supervision réseau et système.

Installation de la solution retenue

L'installation de la solution de supervision retenue s'appuie sur OpenNMS Horizon pour la collecte et l'analyse des métriques, et sur Grafana pour la visualisation avancée des données. Cette section détaille les étapes principales de déploiement et de configuration de ces deux outils, ainsi que la mise en place des tableaux de bord adaptés aux besoins de supervision réseau et système.

Installation d'OpenNMS Horizon

L'installation d'OpenNMS Horizon sur une distribution Debian ou Ubuntu commence par l'ajout du dépôt officiel et de la clé GPG, suivi de la mise à jour des paquets et de l'installation du méta-paquet opennms, qui gère l'ensemble des dépendances nécessaires. Ce processus installe à la fois les services principaux d'OpenNMS (Provisiond, Pollerd, Collectd), l'application web (opennms-webapp-jetty), ainsi que le serveur de base de données PostgreSQL et ses bibliothèques associées. Une fois l'installation terminée, il convient d'initialiser la base de données PostgreSQL, de configurer les accès et de démarrer le service OpenNMS. L'accès à l'interface web permet de finaliser la configuration, notamment en changeant le mot de passe administrateur par défaut.

Pour les environnements nécessitant une installation rapide ou une évaluation, il est également possible d'utiliser un script d'installation rapide ou un déploiement via Docker Compose, qui permet de lancer un système OpenNMS Horizon prêt à l'emploi sur une machine physique ou virtuelle. Ce type de déploiement est particulièrement adapté aux phases de test avant une mise en production à grande échelle.

Installation de Grafana

Grafana, plateforme open source de visualisation de données, s'installe facilement sur Debian ou Ubuntu à partir d'un paquet deb ou via les binaires autonomes. Après téléchargement et installation, il est recommandé de créer un utilisateur système dédié à Grafana pour des raisons de sécurité. Une fois le service démarré, Grafana se connecte à différentes sources de données, dont OpenNMS, via des plugins ou des requêtes personnalisées. L'interface web de Grafana permet ensuite de créer, personnaliser et partager des tableaux de bord dynamiques et interactifs, adaptés à la surveillance des métriques, des logs et des événements issus d'OpenNMS ou d'autres systèmes.

Configuration et création des tableaux de bord

OpenNMS propose plusieurs types de dashboards spécialisés pour répondre à différents besoins de supervision :

- Le dashboard « OpenNMS Basic SNMP Data » offre une vue d'ensemble des métriques SNMP (bande passante, CPU, mémoire) sur les équipements réseau.

- Le dashboard « APC UPS Stats » permet de suivre en détail les onduleurs APC, incluant la charge batterie, la température et la santé des unités.

- Le rapport « OpenNMS Network Interfaces Report » détaille le trafic, les erreurs et les taux d'utilisation des interfaces réseau, facilitant l'identification des points de congestion.

- Le dashboard « OpenNMS World Map Example » localise les équipements sur une carte géographique, utile pour la gestion multi-sites.

- Les dashboards « OpenNMS Outage Dashboard » et « Outage Wallboard » synthétisent les pannes et incidents en cours, offrant une vision claire pour la priorisation des interventions.

- Des dashboards spécifiques existent également pour la supervision d'ActiveMQ, des métriques JVM (Java Virtual Machine) et des performances des nœuds Linux (CPU, mémoire, I/O disques, réseau).

La création des tableaux de bord dans Grafana s'effectue en connectant la plateforme à la source de données OpenNMS, puis en ajoutant des panels (graphiques, jauges, cartes, tableaux) selon les besoins de visualisation. Grafana permet aussi de configurer des alertes personnalisées, envoyées par email, Slack ou autres moyens, en cas de dépassement de seuils critiques.

Bonnes pratiques

Il est conseillé d'utiliser des variables pour rendre les tableaux de bord dynamiques, de limiter le nombre de panels par dashboard pour garantir la lisibilité, et de nommer clairement chaque élément pour faciliter la maintenance. La sauvegarde régulière des

dashboards et la gestion des droits d'accès selon les rôles (lecture seule, édition) sont également recommandées pour assurer la sécurité et la pérennité de la solution2. L'ensemble de ces étapes permet de mettre en place une solution de supervision robuste, évolutive et adaptée aux besoins de l'organisation, en combinant la puissance de collecte et d'analyse d'OpenNMS avec la flexibilité de visualisation de Grafana.

La sécurisation du serveur hébergeant OpenNMS et Grafana repose sur l'application de bonnes pratiques issues de référentiels reconnus comme CIS Benchmarks, NIST SP 800-53 et ISO/IEC 27001. Les premières mesures mises en place incluent la modification immédiate des mots de passe par défaut, la création de comptes utilisateurs à privilèges minimaux, l'activation de mots de passe forts et la suppression des comptes inactifs. Pour Grafana, l'accès est restreint au port 3000 aux seules adresses IP autorisées grâce au pare-feu UFW, le HTTPS est activé via certificat, et la journalisation des accès est configurée. Les ports nécessaires (SSH, OpenNMS Web, Grafana Web, ICMP, SNMP, HTTP) sont autorisés, tandis que tous les autres sont bloqués, avec des règles d'accès restrictives par IP. La journalisation des accès et des tentatives d'authentification est activée, avec des alertes en cas d'échec, et une intégration future avec un IDS/IPS est prévue.

Des recommandations complémentaires renforcent la sécurité : mise à jour régulière et automatisée du système, désactivation des services inutiles, durcissement de la configuration SSH (changement de port, interdiction du root, authentification par clé, installation de fail2ban), mise en place d'un système d'audit (auditd) et de contrôle d'intégrité (AIDE), application stricte du principe du moindre privilège, forçage de l'utilisation de HTTPS avec certificats valides, restriction d'accès par VPN ou filtrage IP, sauvegardes régulières et vérification des procédures de restauration, contrôle des ports réseau ouverts, et supervision de l'intégrité des fichiers critiques. La centralisation des logs est assurée via rsyslog vers une solution SIEM (Grafana Loki, ELK, Wazuh). Pour OpenNMS, la sécurité est renforcée par la modification des identifiants par défaut, l'activation de l'authentification forte, la sécurisation de l'interface web par HTTPS, la restriction de l'accès réseau, l'activation de la journalisation, la mise à jour régulière de la plateforme et la gestion stricte des droits utilisateurs via l'interface d'administration.

Escalade automatique

Dans le cadre de la supervision des infrastructures réseau, l'intégration d'OpenNMS avec Grafana permet de visualiser et de gérer efficacement les alertes critiques. Cependant, un défi majeur réside dans la réactivité des équipes face à ces alertes. Pour y remédier, une procédure de notification automatique a été mise en place dans Grafana. Cette procédure consiste à définir un délai maximal de réponse à une alerte générée par OpenNMS. Si aucune action n'est détectée dans ce délai, une notification est automatiquement envoyée

à un responsable hiérarchique via un canal adapté (webhook, email, messagerie interne, etc.).

La solution repose sur trois éléments : OpenNMS pour la détection des incidents, Grafana pour l'affichage et le suivi des alertes en temps réel, et un système d'automatisation qui surveille le délai de prise en charge. L'implémentation comprend la configuration des alertes dans OpenNMS, leur exposition dans Grafana via un connecteur, la définition de règles d'alerte sur le délai de réponse, et la configuration des notifications d'escalade. Par exemple, une requête Grafana détecte les alertes ouvertes depuis plus de X minutes et déclenche l'envoi d'une notification au superviseur concerné.

Cette approche améliore la réactivité et la continuité du service, tout en assurant que les incidents critiques soient traités rapidement, même en cas d'absence de réaction initiale. Il est recommandé de choisir un délai adapté à la criticité des alertes, de tester régulièrement le système et de documenter le processus pour tous les acteurs impliqués.

Alerte

La supervision réseau avec OpenNMS Horizon permet de surveiller en temps réel l'état de santé des équipements et services informatiques d'une organisation. Un élément central de cette supervision est la configuration d'alertes efficaces pour détecter rapidement les incidents, qu'il s'agisse de pannes, de dégradations de performance ou de problèmes de sécurité, et pour alerter les bonnes personnes au bon moment.

Le système repose sur plusieurs concepts clés :

- Événements (UEI) : Chaque incident détecté génère un événement unique, facilitant le suivi et la gestion des alertes.

- Seuils de performance (Thresholds) : Des seuils sont définis sur des indicateurs comme l'utilisation du CPU, de la mémoire, du disque ou du trafic réseau. Lorsqu'une valeur critique est atteinte, un événement est déclenché.

- Sondes de disponibilité (Pollers) : OpenNMS vérifie régulièrement la disponibilité des services (ping, HTTP, SSH, etc.) et génère une alerte si un service devient indisponible après plusieurs tentatives.

- Notifications : Lorsqu'un événement critique survient, des notifications automatiques (email, SMS, etc.) sont envoyées aux destinataires appropriés, selon des règles personnalisées.

- La configuration des seuils peut se faire via l'interface web ou par modification des fichiers XML, permettant d'adapter la supervision aux besoins spécifiques de l'organisation. Il est possible de tester et valider le bon fonctionnement des alertes, soit en provoquant la condition réelle, soit en simulant un événement. L'association des seuils aux ressources surveillées se fait par groupes et filtres, assurant une application ciblée et efficace.

La mise en place de cette supervision automatisée optimise la détection des incidents, améliore la réactivité des équipes et contribue à la continuité de service, tout en offrant une grande flexibilité dans la gestion des alertes et des notifications.

Notification

OpenNMS offre un système de notifications performant permettant d'alerter automatiquement les utilisateurs lorsqu'un événement spécifique survient sur le réseau, comme une panne de service ou le dépassement d'un seuil. Chaque événement est identifié par un identifiant unique (UEI), ce qui permet de cibler précisément les notifications. Celles-ci sont déclenchées par ces UEI et peuvent être personnalisées pour être envoyées à différents utilisateurs ou groupes, selon le type et la gravité de l'événement.

Le système prend en charge plusieurs méthodes d'envoi, telles que l'e-mail, le SMS ou l'intégration avec des plateformes comme Slack, en configurant des commandes et des chemins de notification qui définissent les destinataires et le mode de diffusion.

Cette approche flexible permet d'adapter les alertes aux besoins opérationnels de l'organisation, garantissant que les incidents critiques sont communiqués rapidement aux bonnes personnes. Les notifications peuvent également être escaladées à d'autres contacts si l'événement n'est pas reconnu, ce qui favorise une gestion efficace des incidents. Les administrateurs peuvent personnaliser le contenu des notifications (sujet, corps du message) et utiliser des groupes de contacts pour une gestion simplifiée. Il est recommandé de tester régulièrement le système et d'activer les fonctions d'accusé de réception et d'escalade pour assurer la fiabilité et l'efficacité des notifications

Service TIC

Le service TIC (Technologies de l'Information et de la Communication) d'OpenNMS joue un rôle central dans la surveillance, la gestion et l'optimisation des réseaux informatiques, tant pour les infrastructures internes que pour les services destinés au public. OpenNMS est une

plateforme open source capable de surveiller des milliers d'équipements et de services en temps réel, en utilisant des protocoles standards comme SNMP, ICMP, ou encore des flux NetFlow et sFlow. Grâce à la découverte automatique, la plateforme identifie et inventorie les équipements et services présents sur le réseau, facilitant ainsi la gestion des ressources et la détection rapide des nouveaux dispositifs.

Le fonctionnement d'OpenNMS repose sur la collecte et l'analyse de données de performance (comme la bande passante, la disponibilité des services, ou la charge CPU), la génération d'événements et d'alarmes en cas d'incident, et la possibilité d'envoyer des notifications personnalisées aux équipes concernées. Les utilisateurs bénéficient d'une interface web intuitive qui offre une vue d'ensemble de l'état du réseau, des événements en cours, des alarmes, et des statistiques détaillées. Pour le public ou les utilisateurs finaux, cela se traduit par une meilleure disponibilité des services numériques, une détection proactive des pannes, et une capacité à informer rapidement sur l'état des services critiques. OpenNMS permet aussi de configurer des seuils d'alerte pour anticiper les problèmes avant qu'ils n'affectent les utilisateurs, et d'intégrer des cartes topologiques pour visualiser les relations entre les différents éléments du réseau.

En résumé, le service TIC basé sur OpenNMS aide les organisations à garantir la fiabilité, la performance et la sécurité de leurs services numériques, tout en offrant des outils puissants pour la supervision, l'analyse et la communication avec le public en cas d'incident. Sa flexibilité et son évolutivité en font une solution adaptée aussi bien aux petites qu'aux grandes structures, permettant de surveiller efficacement des réseaux complexes et distribués

Conclusion