Planifier le déploiement

Un système de supervision doit répondre aux enjeux attendus. La 1ère étape indispensable est d'obtenir l'accord de déploiement en mode projet afin de définir en détail toutes les étapes et contraintes, et notamment :

- Numbre de checks à superviser
- Fréquence des checks
- Méthode de supervision des checks Passif versus Actif
- Protocoles utilisés pour l'acquisition de données (ping, SSH, NSCA, TSCA, SNMP, NRPE, NSCAweb, collectd, scripts, etc)
- Durée de rétention des données de performance
- Pour chaque statut ou donnée de performance, définir les exigences à atteindre .

Niveau de scalabilité de Shinken Enterprise

Shinken Enterprise peut être scalé horizontalement et verticalement . Il va gérer automatiquement la rétention distribuée de statuts. Il n'est pas nécessaire de rajouter un cluster externe ou une architecture en HA. .

La scalabilité peut se décrire à travers plusieurs points

- Nombre d'hôtes et de checks supervisés
- Nombre de checks actifs par seconde (le type a également un impact énorme t!)
- Nombre de résultats de checks par seconde

Et à un niveau moindre, les données de performance ne devant pas surcharger le serveur graphite

Passif versus Actif

Les checks passifs n'ont pas besoin d'être planifiés sur le serveur de supervision. L'acquisition de donnés et le processing sont distribués aux hôtes supervisés permettant de réduire les intervalles d'acquisition en augmentant les points de collecte.

Les checks actifs bénéficient des algorithmes puissants de Shinken Enterprise.

Une installation importante doit bénéficier des 2 types de checks.

Scalier le broker

Le broker est un élément clé de la scalabilité de l'architecture. Un seul broker peut être actif par scheduler. Un broker peut processer des messages de plusieurs schedulers. Dans la plupart des déploiements, Livestatus est le module broker qui met à disposition les informations de statut aux frontends web (Nagvis, Multisite, Thruk, etc.) ou à l'interface graphique intégrée à Shinken Enterprise. Le broker a besoin de mémoire et de puissance de calcul.

Modèle de dépendance

Shinken Enterprise a un grand modèle de gestion des dépendances. La détection automatique de problème source au niveau de l'hôte est une des méthodes proposée par Shinken Enterprises. Cela se fait à travers la définition claire des liens parents/enfants. Cela signifie que sur un problème, le système va automatiquement replanifier une vérification du ou des parents définis. Une fois que le problème source est identifié, tous les enfants seront marqués en statut inconnu..

Ce modèle est très pratique pour réduire les faux positifs. Ce qui est important, c'est la définition de l'arbre de dépendances. Cet arbre est limité à un seul scheduler.

Splitting trees by a logical grouping makes sense. This could be groups of checks, geographic location, network hierarchy or other. Some elements may need to be duplicated at a host level (ex. ping check) comme certains éléments critiques (core routers, datacenter routers, AD, DNS, DHCP, NTP, etc.). Un arbre typique va inclure des clients, serveurs, chemin réseau, et checks associés .

Scaler les démons d'acquisition

Typiquement, les pollers et schedulers utilisent le plus de ressources réseau, mémoire et CPU . Utilisez l'architecture distribuée pour scaler horizontalement sur plusieurs serveurs basiques. Utilisez au moins une paire de Scheduler sur chaque serveur . Votre modèle doit permettre au moins 2 arbres, idéalement 4..

Méthode d'acquisition passive

Les données de performance ou métrics (mode Nagios) sont intégrés dans les retours de checks . Un retour pour n'avoir aucun ou plusieurs metrics associés.
Ceux-ci sont transférés de manière transparente aux systèmes externes via le Broker. Le broker Graphite peut envoyer plus de 2000 metrics par seconde. Graphite peut être configuré pour atteindre plus de 80K metrics par seconde.

Si un metric n'a pas besoin de son propre check, il doit être associé à un check de nature similaire . Les checks sont les composants les plus "chers", Vous ne devriez pas excéder les **60K checks* pour des performances optimisées.

Protocoles recommandés pour l'acquisition passive

- Ws_Arbiter (Used by GLPI)
- NSCA (generic collection)

Méthode de gestion des Log

Les logs système et réseau doivent être récoltées depuis les serveurs et équipements réseau. Pour cela, un système centralisé est requis :.

Nous suggérons les systèmes suivants:

- OSSEC+Splunk
- loglogic