08 - Supervision avec Nagios

Objectif

- Présenter la supervision comme un système d'observation, d'alerte et parfois d'action sur l'infrastructure.

1. Définition de la supervision

> La supervision sert à visualiser, surveiller et parfois agir.

1. Définition de la supervision

- La supervision regroupe des outils et pratiques destinés à suivre l'état technique d'un réseau, de serveurs, de services et d'applications.
- Pour l'entreprise, l'objectif est la disponibilité.
- Pour l'administrateur, c'est la capacité à voir l'état du système, prévenir les pannes, identifier rapidement un incident et mesurer son impact.

2. Définir le périmètre

> Avant de choisir un outil, il faut savoir ce que l'on veut obtenir.

2. Définir le périmètre — Trois intentions

- **Visualiser** : comprendre l'état de santé et les dépendances.
- **Surveiller** : détecter une anomalie et alerter.
- **Agir** : déclencher une réponse ou une remédiation.

2. Définir le périmètre

- Une supervision sans périmètre devient vite un mur d'alertes.
- Il faut décider ce qui est critique, ce qui est informatif, qui reçoit quoi, et dans quel délai.

3. Visualiser l'architecture

> La cartographie permet de comprendre les impacts.

3. Visualiser l'architecture

- Une cartographie de supervision aide à voir les points sensibles, les dépendances, les goulets d'étranglement et l'effet d'une intervention.
- Elle permet aussi de prévoir l'évolution de l'architecture : débit d'un lien, charge d'un serveur, saturation d'un stockage.

4. Contrôles actifs et passifs

> En actif, le superviseur interroge ; en passif, l'équipement ou un collecteur envoie l'information.

4. Contrôles actifs et passifs — Contrôle actif

- Nagios exécute un plugin pour vérifier un hôte ou un service.
- Le plugin retourne un état.

4. Contrôles actifs et passifs — Contrôle passif

- Une application externe, un agent ou un collecteur envoie un résultat vers Nagios.
- La charge côté superviseur peut être plus faible.

4. Contrôles actifs et passifs — Point d'attention

- Dans les deux cas, Nagios analyse ensuite les résultats, applique les seuils, déclenche les états, notifications et éventuels gestionnaires d'événements.

5. Domaines supervisés

> On ne supervise pas seulement des machines, on supervise des services rendus.

5. Domaines supervisés — Catégories

- supervision réseau et matérielle ;
- supervision système ;
- supervision services et applications.

5. Domaines supervisés — Indicateurs vitaux

```
État des liens et ports : up, down, disabled, blocked
Débit des liens : détection des goulets d'étranglement
CPU : charge processeur
Disque : espace libre et I/O
Ventilation/température : santé matérielle
Mémoire : pression mémoire
Services : disponibilité et temps de réponse
```

6. Nagios

> Nagios repose sur un cœur de supervision, des plugins et des objets configurés.

6. Nagios

- Nagios, anciennement NetSaint, est une solution libre de supervision.
- Son principe fort est simple : un plugin teste quelque chose et retourne un code d'état que Nagios interprète.

6. Nagios — Fonctions importantes

- suivi de ressources hôte : CPU, disque, logs ;
- surveillance de services réseau ;
- scripts personnalisés ;
- contrôles distants via SSH, SSL, NRPE ou autres mécanismes ;
- notifications ;
- hiérarchie parent/enfant pour distinguer panne et inaccessibilité ;
- interface web ;
- stockage souvent basé sur fichiers texte dans la configuration classique.

7. Architecture générale

> Nagios orchestre des contrôles, mais ce sont les plugins qui savent tester.

7. Architecture générale — Composants

- **Serveur Nagios** : planifie, reçoit, interprète, alerte.
- **Plugins** : exécutent les tests.
- **Stockage** : conserve états, historiques et configuration.
- **Objets distants** : hôtes, services, équipements.
- **Interface web** : visualisation et exploitation.

8. Plugins Nagios

> Un plugin Nagios est un programme qui retourne un code standardisé.

8. Plugins Nagios — Codes de retour

```
0 OK       le service fonctionne
1 WARNING  le service est dégradé
2 CRITICAL le service ne fonctionne plus
3 UNKNOWN  l'état n'est pas déterminable
```

8. Plugins Nagios — Plugins courants

```
check_disk   espace disque
check_http   service HTTP
check_mysql  base MySQL
check_nt     informations Windows
check_nrpe   checks distants Linux/Unix
check_ping   présence et latence
check_pop    service POP
check_snmp   informations via SNMP
```

9. NRPE

> NRPE permet à Nagios de déclencher des checks locaux sur une machine distante Linux/Unix.

9. NRPE

- Certains contrôles doivent s'exécuter localement sur la machine distante : espace d'une partition, nombre de processus, charge CPU, processus zombies.
- Nagios appelle `check_nrpe`, qui demande au service NRPE distant d'exécuter une commande autorisée.

10. Fichiers de configuration Nagios

> Nagios est très déclaratif : on décrit des hôtes, services, contacts, commandes et périodes.

10. Fichiers de configuration Nagios — Emplacement type

```
/opt/nagios/etc/
```

10. Fichiers de configuration Nagios — Fichiers courants

```
cgi.cfg                  interface web
nagios.cfg               configuration principale
objects/commands.cfg     commandes de check
objects/contacts.cfg     contacts et groupes
objects/localhost.cfg    supervision du serveur Nagios
objects/templates.cfg    modèles
objects/timeperiods.cfg  périodes horaires
objects/hostclients.cfg  clients à surveiller
objects/hostservers.cfg  serveurs à surveiller
objects/network.cfg      équipements réseau
```

11. Templates

> Les templates évitent de répéter les mêmes paramètres sur chaque hôte ou service.

11. Templates — Exemple simplifié

```
define host{
  name                    generic-host
  check_command           check-host-alive
  check_period            24x7
  max_check_attempts      10
  contact_groups          admins
  notification_options    d,u,r
  register                0
}
```

- `register 0` indique que l'objet est un modèle et non un hôte réel à superviser.

11. Templates — Spécialiser un template

```
define host{
  use                     generic-host
  name                    tmpl-host-web
  check_period            24x7
  contact_groups          support
  register                0
}
```

11. Templates — Déclarer un hôte réel

```
define host{
  use             tmpl-host-web
  host_name       Rainette
  alias           Serveur Web Rainette
  address         xx.xx.xx.xx
  contact_groups  support,admins
}
```

11. Templates — Déclarer un service

```
define service{
  use                   generic-service
  host_name             Rainette
  service_description   Reponse interface Web Nagios
  check_command         check_http!"http://xx.xx.xx.xx/nagios"
}
```

12. Hostgroups et services groupés

> Un hostgroup permet d'appliquer un service à plusieurs machines sans dupliquer la configuration.

```
define hostgroup {
  hostgroup_name  Serveur_web
  alias           Groupe des Serveurs WEB
  members         serveur1,serveur2
}
```

> Service appliqué au groupe :

```
define service{
  use                   generic-service
  hostgroup_name        Serveur_web
  service_description   Reponse HTTP
  check_command         check_http
}
```

13. Mise en place NRPE

13. Mise en place NRPE — Côté serveur Nagios

```
apt-get install nagios-nrpe-plugin
```

- Le fichier de configuration des plugins peut fournir des commandes comme `check_nrpe` et `check_nrpe_1arg`.

13. Mise en place NRPE — Côté machine distante

- Modifier `/etc/nagios/nrpe.cfg` :

```
allowed_hosts=<IP_du_serveur_Nagios>
```

- Redémarrer le service :

```
/etc/init.d/nagios-nrpe-server start
```

- Définir une commande distante :

```
command[check_sda1]=/usr/lib/nagios/plugins/check_disk -w 20 -c 10 -p /dev/sda1
```

13. Mise en place NRPE — Déclarer le service côté Nagios

```
define service{
  use                  generic-service
  host_name            hera
  service_description  disk check
  check_command        check_nrpe_1arg!check_sda1
}
```

Conclusion orale

- La supervision n'est pas seulement un tableau vert ou rouge.
- C'est une méthode pour transformer des signaux techniques en décisions opérationnelles : alerter, diagnostiquer, prioriser et parfois agir automatiquement.