

Administration et configuration d'un outil de supervision ZABBIX sous linux



Département des Mathématiques et Informatique: Réseaux et Cloud Computing

Réalisé par: Sara BEN CHIKH
Safaa HALLI

Encadrant: Nouredine ABGHOUB
Khalid MOUSSAID

Année académique : 2020/2021

Tableau des contenus

Remerciement	4
Dedicace	5
Introduction générale	6
1. ZABBIX	8
1.1. Historique de zabbix.....	8
1.2. Définition de Zabbix.....	8
1.3. Qu'est-ce que Zabbix ?	9
1.4. Fonctionnalités Zabbix.....	9
1.4.1. Collecte de données.....	9
1.4.2. Définitions de seuils flexibles.....	9
1.4.3. Alertes hautement personnalisables.....	10
1.4.4. Graphiques en temps réel.....	10
1.4.5. Capacités de supervision Web.....	10
1.4.6. Options de visualisation étendues.....	10
1.4.7. Stockage de données historiques.....	10
1.4.8. Configuration facile.....	10
1.4.9. Utilisation de modèles.....	10
1.4.10. Découverte du réseau.....	10
1.4.11. Interface Web rapide.....	10
1.4.12. API Zabbix.....	10
1.4.13. Système de permissions.....	10
1.4.14. Agent complet et facilement extensible.....	11
1.4.15. Démons binaires.....	11
1.4.16. Prêt pour les environnements complexes.....	11
2. NODE.JS	11
2.1. Historique	11
2.2. Définition	11
2.3. Utilisation courante	12
2.4. Frameworks pour Node.js	12
2.5. Outils pour Node.js	13
3. Tomcat	13
3.1. Définition	13
3.2. Environnement	13
3.3.État de développement.....	14
3.4. Principe de fonctionnement	14
3.5. Arborescence de répertoires.....	15
3.6. Historique	15
3.7. Fonctionnalités des versions	16
3.8. Organisation des répertoires de Tomcat	17
3.8.1. Répertoire bin.....	17

3.8.2. Répertoire conf.....	17
3.8.3. Répertoire lib.....	17
3.8.4. Répertoire log.....	17
3.8.5. Répertoire temp.....	17
3.8.6. Répertoire webapp.....	18
3.8.7. Répertoire work.....	18
4. Active Directory.....	18
4.1. Caractéristiques d'Active Directory.....	18.
4.2. Principe de fonctionnement d'Active Directory.....	18
4.3. Structure Active Directory.....	18
4.4. Les avantages des services de domaine Active Directory.....	19
4.5. Services de domaine Active Directory (AD DS) : les termes à connaître	20
4.6. Quels sont les services fournis par AD DS ?.....	20
5. MARIADB.....	21
5.1. Présentation.....	21
5.2. Historique.....	21
5.3. Sécurité.....	21
5.4. Compatibilité.....	22
5.5. Versions actuelles et Calendrier des sorties.....	22
5.6. Exécution et gestion des licences MariaDB.....	22
6. MYSQL.....	23
6.1. Définition	23
6.2. Historique	23
6.3. Dénomination	24
6.4. Utilisation	24
6.5. Caractéristiques	25
6.5.1. Systèmes d'exploitation supportés	25
6.5.2. Langages informatiques supportés	25
6. Version de SQL.....	25
7. Fonctionnalités	26
8. Interopérabilité.....	26
9. Usage en production.....;	26
Conclusion général.....	27
Webographie.....	28

Remerciements

Nous tenons à exprimer toute notre reconnaissance à nos professeurs encadrants de mémoire, Mr. Nouredine ABG HOUR et Mr. Khalid MOUSSAID. Nous les remercions de nous avoir encadrées, orientées, aidées, et conseillées.

Nous adressons nos sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé nos réflexions et ont répondu à nos questions durant nos recherches.

Nous remercions nos très chers parents, qui ont toujours été là pour nous. Nous remercions nos sœurs et nos frères, pour leurs encouragements.

Enfin, nous remercions nos amis pour leur soutien inconditionnel. Leurs encouragements ont été d'une grande aide.

À tous ces intervenants, nous présentons nos remerciements, notre respect et gratitude.

Dédicace

Tout d'abord nous avons le plaisir d'offrir ce rapport à nos familles et surtout nos parents qui n'ont jamais cessé de nous aimer, de veiller sur notre bien-être, et de nous soutenir moralement et financièrement.

Nous dédions ce travail à nos professeurs Mr Nouredine ABG HOUR, Mr Khalid MOUSSAID, Mr. Mohamed MOUGHIT et Mme. Amina LOMARI.

Enfin à tous nos amis et à ceux qui nous ont aidées à réaliser ce travail.

Introduction générale

Les entreprises quel que soit leur domaine veillent toujours à être dans le centre de la concurrence économique et à garder leur bonne réputation, et pour cela elles donnent beaucoup d'importance à leur système informatique avec toutes ses composantes, parce qu'il garantit le bon fonctionnement de leurs différentes activités.

Une erreur ou une panne influe sur la performance du réseau et sur l'entreprise en général. La solution pour les éviter c'est de superviser le système entier, pour permettre à l'administrateur de l'analyser et de le gérer en tout temps.

La supervision permet la surveillance du bon fonctionnement des systèmes d'informations. Elle permet aux administrateurs réseau de surveiller les différentes composants matériels et logiciels, les visualiser, et analyser les différentes informations et données fournies sur eux. L'administrateur peut donc vérifier le fonctionnement normal ou anormal du système informatique et agir pour résoudre ses problèmes.

L'objectif de la supervision est la visibilité de toutes les composantes du réseau pour l'administrateur, afin de lui permettre de contrôler l'infrastructure informatique et de la gérer plus facilement surtout avec l'évolution du réseau. Cela garantit la fiabilité des différents services des entreprises et des administrations.

La supervision fournira des rapports sur le fonctionnement du système comme l'utilisation du CPU, l'occupation de la mémoire physique, l'espace libre des disques durs, la fluctuation de la température,...etc. Ces rapports permettront à l'administrateur d'anticiper les incidents sur le parc informatique, de piloter son réseau et de réaliser des actions en fonction des alertes déclenchées. La supervision facilite la tâche de l'administrateur et garantit le bon fonctionnement des serveurs. Les outils de supervision vont nous permettre de visualiser l'état de santé de notre parc, ce qui est très souvent apprécié par les entités dirigeantes des entreprises.

Deux phases sont importantes pour que les administrateurs soient capables d'atteindre l'objectif voulu par la supervision, à savoir, surveiller le système et garantir sa disponibilité même en cas d'anomalie.

- Tenter de prévenir en cas de problème (défaillances matérielles ou interruption des services) et garantir une remontée d'information rapide ;
- Automatiser les tâches de récupération des applications et des services en assurant des mécanismes de redondance en une durée d'intervention minimale (par exemple : le redémarrage des services interrompus, l'arrêt de la machine en cas de surcharge du CPU, la sauvegarde des données en cas de risque de perte d'un disque dur en miroir, etc.).

Une grande majorité des logiciels de supervision sont basés sur le protocole SNMP qui existe depuis de nombreuses années. La plupart de ces outils permettent de nombreuses fonctions dont voici les principales:

- Surveillance du réseau.
- Visualisation des composantes du système.
- Analyser les problèmes.
- Déclencher des alertes en cas de problèmes.
- Effectuer des actions en fonction des alertes.

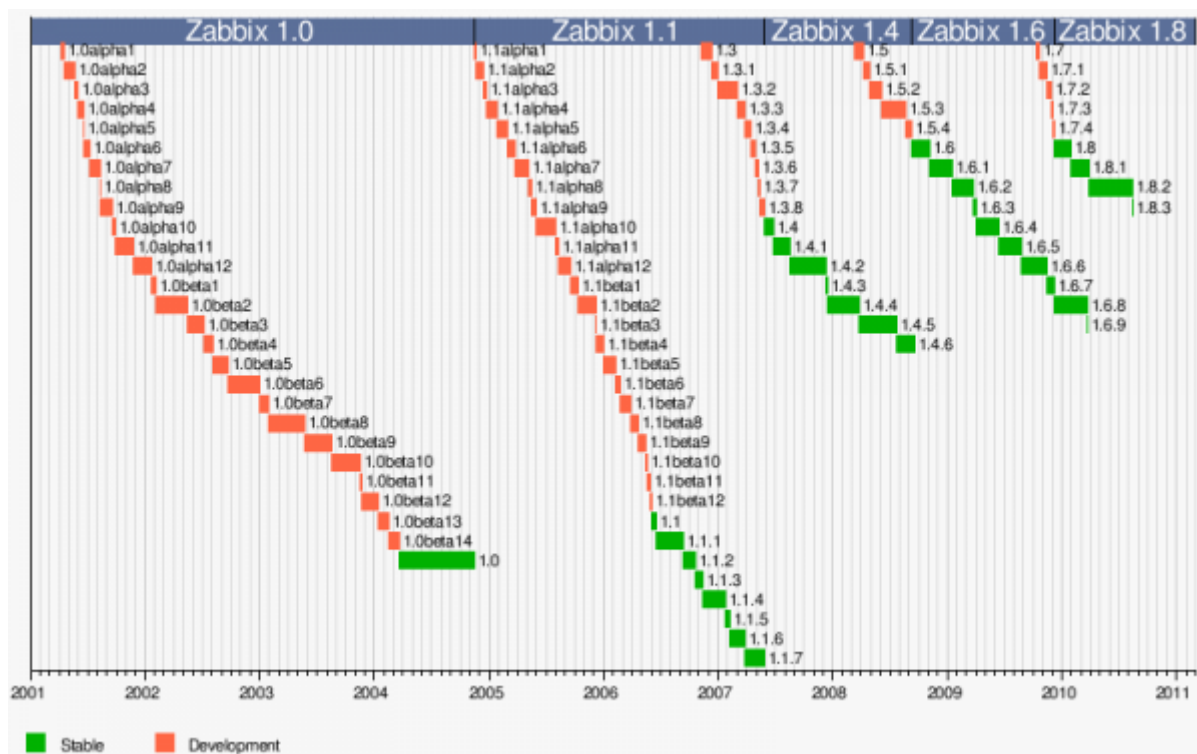
Le travail de l'administrateur est alors simplifié. Les outils de supervision lui donnent un schéma généralisé du système d'information pour surveiller ses différentes fonctionnalités en temps réel, et agir pro-activement.

1. ZABBIX

1.1. Historique de zabbix

Au commencement, Zabbix vu le jour en 1998 développé par Alexei Vladishev, sous la forme d'un projet interne, afin de répondre aux besoins de supervision d'une banque. C'est seulement à partir de 2001 que le logiciel passa sous licence GPL, avec la sortie d'une première version alpha de Zabbix 1.0. Après plusieurs années de développement, la version finale 1.0 fut disponible le 23 mars 2004.

Mise à part la version 1.1, qui est sortie environ 2 ans plus tard la première, l'évolution de Zabbix est constante, avec en moyenne 2 sorties majeures par an pour arriver aujourd'hui à la version 1.8, sortie en décembre 2009. Zabbix est actuellement disponible sous la version 1.8.3.



De nos jours, Zabbix est développé sous licence GPL v2 par l'entreprise Zabbix SIA, fondée en 2005, afin de fournir un support professionnel et des services d'intégration, de déploiement, de consulting, ou encore de formation, ainsi que différents partenariats avec des entreprises à l'international.

Avec un nombre toujours croissant d'utilisateurs dans les infrastructures IT, Zabbix est devenu en l'espace de quelques années, l'une des applications open source de supervision les plus populaires dans le monde.

1.2. Définition de Zabbix

Zabbix est un logiciel gratuit qui supervise de nombreux paramètres d'un réseau ainsi que la santé et l'intégrité des serveurs. Zabbix utilise un mécanisme de notification flexible qui permet aux utilisateurs de configurer un alerte e-mail pour pratiquement tous les événements. Cela permet une réponse rapide aux problèmes serveurs. Zabbix offre un excellent reporting et des fonctionnalités de

visualisation de données basées sur les données stockées. Cela rend Zabbix idéal pour la gestion de la capacité.

Zabbix supporte à la fois l'interrogation (polling) et la réception (trapping) de données. Tous les rapports et statistiques de Zabbix, comme la configuration de paramètres, sont accessibles par l'interface web. L'interface web veille à ce que le statut de votre réseau et de vos serveurs puisse être évalué depuis n'importe quel endroit. Correctement configuré, Zabbix peut jouer un rôle important dans la supervision de l'infrastructure IT. Ceci est également vrai pour les petites organisations avec peu de serveurs ainsi que pour les grandes entreprises avec une multitude de serveurs.

Zabbix est écrit et distribué sous licence GPL (General Public License) version 2. Cela signifie que son code source est librement distribué et disponible pour le public.

1.3. Qu'est-ce que Zabbix ?

Zabbix est une solution complète de supervision libre et open-source, sous licence GPLv2.

La particularité de Zabbix par rapport à d'autre produit comme Nagios ou autres c'est qu'il fonctionne sur la base client/serveur, avec un agent installé sur les hôtes supervisés.

De plus, Zabbix est composé de trois parties :

- **zabbix-server-mysql** : le serveur proprement dit.
- **zabbix-frontend-php**: interface web.
- **zabbix-agent** : l'agent de supervision installé sur les hôtes.

Sur le serveur, ZABBIX utilise MySQL, PostgreSQL ou Oracle pour stocker les données. Ici nous utiliserons MySQL.

Le serveur de données doit être un UNIX.

Un agent ZABBIX peut être installé sur les hôtes Linux, UNIX et Windows.

Son interface web est écrite en PHP.

1.4. Fonctionnalités Zabbix

Zabbix est une solution de supervision de réseaux hautement intégrée, offrant une multitude de fonctionnalités dans un seul package.

1.4.1. Collecte de données

- disponibilité et contrôles de performance.
- prise en charge SNMP (réception et interrogation), IPMI, JMX, surveillance VMware.
- vérifications personnalisées.
- collecte des données souhaitées à des intervalles personnalisés
- effectuée par le serveur/proxy et par des agents.

1.4.2. Définitions de seuils flexibles

- vous pouvez définir des seuils de problèmes très flexibles, appelés déclencheurs, référençant des valeurs à partir de la base de données principale.

1.4.3. Alertes hautement personnalisables

- l'envoi de notifications peut être personnalisé selon le calendrier d'escalade, le destinataire ou le type de média.
- les notifications peuvent être significatives et utiles en utilisant des variables macro.
- les actions automatiques incluent des commandes à distance.

1.4.4. Graphiques en temps réel

- les éléments supervisés sont immédiatement représentés graphiquement à l'aide de la fonctionnalité graphique intégrée.

1.4.5. Capacités de supervision Web

- Zabbix peut suivre un chemin de clics de souris simulés sur un site Web et ainsi vérifier le bon fonctionnement et le temps de réponse.

1.4.6. Options de visualisation étendues

- possibilité de créer des graphiques personnalisés pouvant combiner plusieurs éléments dans une même vue.
- représentation du réseau sous forme de carte.
- des écrans personnalisés et des diaporamas pour un aperçu dans un tableau de bord.
- des rapports.
- vue de haut niveau (métier) des ressources surveillées.

1.4.7. Stockage de données historiques

- données stockées dans une base de données.
- historique configurable.
- procédure de nettoyage intégrée.

1.4.8. Configuration facile

- ajouter des équipements surveillés en tant qu'hôtes.
- les hôtes sont pris en charge pour la surveillance, une fois dans la base de données.
- appliquer des modèles aux équipements surveillés.

1.4.9. Utilisation de modèles

- grouper les vérifications dans des modèles.
- les modèles peuvent hériter d'autres modèles.

1.4.10. Découverte du réseau

- découverte automatique des équipements réseaux.
- enregistrement automatique de l'agent.
- découverte de systèmes de fichiers, d'interfaces réseau et d'OID SNMP.

1.4.11. Interface Web rapide

- une interface web en PHP.
- accessible de n'importe où.
- la navigation se fait via des clics.
- journal d'audit.

1.4.12. API Zabbix

- L'API Zabbix fournit une interface programmable pour des manipulations de masse sur Zabbix, l'intégration de logiciels tiers et à d'autres fins.

1.4.13. Système de permissions

- authentification de l'utilisateur sécurisée.

- certains utilisateurs peuvent être limités à certaines vues.

1.4.14. Agent complet et facilement extensible

- déployés sur des cibles de surveillance.
- peut être déployé sur Linux et Windows.

1.4.15. Démons binaires

- écrit en C, pour des performances et une faible empreinte mémoire.
- facilement portable/

1.4.16. Prêt pour les environnements complexes

- supervision à distance facilitée en utilisant un proxy Zabbix.

2. Node.js

2.1. Historique:

Node.js a été créé par Ryan Dahl en 2009. Son développement et sa maintenance sont effectués par l'entreprise Joyent (en). Dahl a eu l'idée de créer *Node.js* après avoir observé la barre de progression d'un chargement de fichier sous Flickr : le navigateur ne savait pas quel pourcentage du fichier était chargé et devait adresser une requête au serveur web. Dahl voulait développer une méthode plus simple¹⁴. Le serveur web Mongrel de Ruby a été l'autre source d'inspiration pour Dahl¹⁵. Dahl avait échoué dans plusieurs projets en C, Lua et Haskell, mais, quand le moteur V8 fut diffusé, il commença à s'intéresser à JavaScript. Même si son idée d'origine était d'utiliser des entrées et sorties non bloquantes, il en reviendra un peu dans son module système car cela provoquait des problèmes de chargement de bibliothèques externes¹⁷.

Node.js est utilisé au cœur du système d'exploitation Palm webOS.

2.2. Définition:

Node.js est une plateforme logicielle libre en JavaScript, orientée vers les applications réseau évènementielles hautement concurrentes qui doivent pouvoir monter en charge.

Elle utilise la machine virtuelle V8, la librairie libuv pour sa boucle d'évènements, et implémente sous licence MIT les spécifications CommonJS.

Parmi les modules natifs de Node.js, on retrouve http qui permet le développement de serveur HTTP. Ce qui autorise, lors du déploiement de sites internet et d'applications web développés avec Node.js, de ne pas installer et utiliser des serveurs webs tels que Nginx ou Apache.

Concrètement, Node.js est un environnement bas niveau permettant l'exécution de JavaScript côté serveur.

Node.js est utilisé notamment comme plateforme de serveur Web, elle est utilisée par GoDaddy, IBM, NetFlix, Amazon Web Services, Groupon, Vivaldi, SAP, LinkedIn , Microsoft , Yahoo!, Walmart, Rakuten, Sage et PayPal.

2.3. Utilisation courante :

Node.js est utilisé pour faire des applications cross-plateforme avec des framework comme Ionic pour les applications mobiles ou encore Electron pour les applications de bureau. Les géants comme Discord ou encore Slack utilisent ce système.

Node.js est aussi beaucoup utilisé pour faire des serveurs de bot informatique ou bien des serveurs web. Certaines API Rest pour l'authentification sont faites avec Node.js.

2.4. Frameworks pour Node.js :

- Adonis
- AllcountJS
- Catberry
- Compound.js
- Connect
- DerbyJS
- Devis
- diet.js
- Express
- Feathers
- Flatiron
- FoalTS
- Fortune.js
- Frisby
- Geddy
- Hapi.js
- Ionic
- Keystone
- Knockout.js
- Koa.js
- kraken.js
- Locomotive
- Mean.js et Mean.io
- Meteor
- Mojito
- Nest.js
- Nodal
- Partial.js
- percolator
- Raddish
- RESTify
- RhapsodyJS
- SailsJS
- seneca.js
- Socket.IO
- SocketStream
- Strapi
- ThinkJS
- TotalJS
- TowerJS
- Trails

2.5. Outils pour Node.js :

nCombo : framework pour le cloud.

TestCafé: outil d'automatisation de tests d'applications web.

Browserify : outil pour utiliser les modules Node.js dans le navigateur.

Babel: Compilateur JavaScript open-source.

Mongoose : couche de persistance pour MongoDB.

Grunt : outil d'automatisation des tâches.

PM2 : outil de surveillance et d'automatisation de tâches en cas de crash d'une application Node.js.

memwatch-next : un paquet fourni par Node Package Manager (npm) pour la surveillance de l'utilisation de mémoire. Utile pour détecter les fuites de mémoires lentes.

Gulp : outil d'automatisation des tâches (basé sur les streams).

Bower: gestionnaire de paquet pour le web.

Yeoman : générateur de site web utilisant Grunt et Bower.

Unit.js : framework pour l'écriture des tests unitaires.

Mocha : runner pour exécuter les tests unitaires.

Jasmine : tests unitaires

3. Tomcat

3.1. Définition :

Apache Tomcat ou simplement Tomcat est un serveur d'applications, plus précisément un conteneur web libre de servlets et JSP. Issu du projet Jakarta, c'est un des nombreux projets de l'Apache Software Foundation. Il implémente les spécifications des servlets et des JSP du Java Community Process9, est paramétrable par des fichiers XML et des propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.

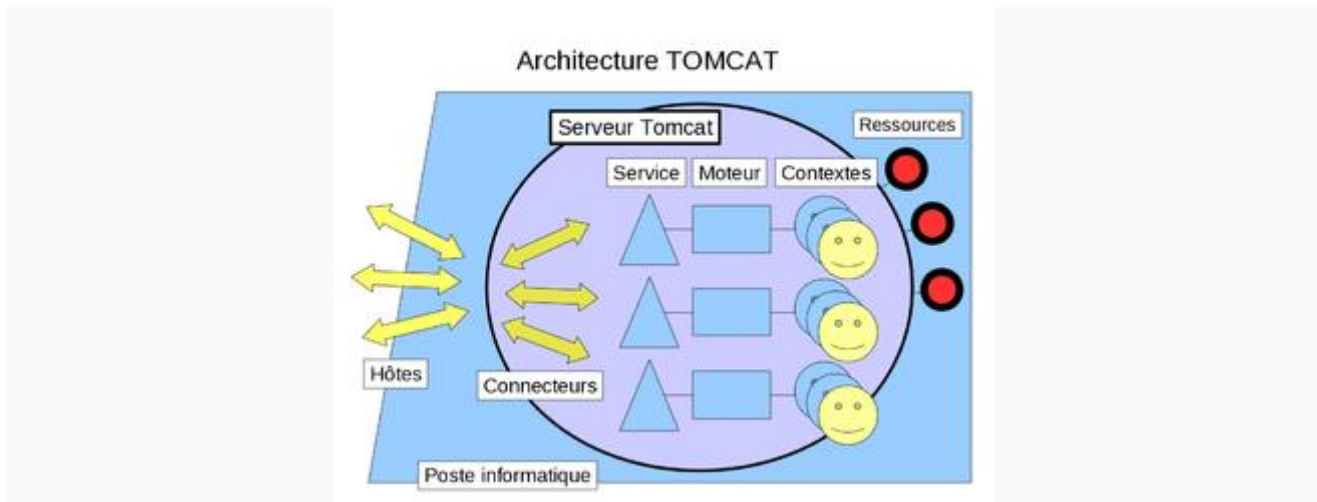
3.2. Environnement :

Tomcat est un serveur HTTP à part entière. De plus, il gère les servlets et les JSP (par un compilateur Jasper compilant les pages JSP pour en faire des servlets). Tomcat a été écrit en langage Java. Il peut donc s'exécuter via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant.

Catalina est le conteneur de servlets utilisé par Tomcat. Il est conforme aux spécifications servlet d'Oracle Corporation et les JavaServer Pages (JSP). Coyote est le connecteur HTTP de Tomcat,

compatible avec le protocole HTTP 1.1 pour le serveur web ou conteneur d'application. Jasper est le moteur JSP d'Apache Tomcat. Tomcat 9.x utilise Jasper 2, qui est une implémentation de la spécification JavaServer Pages 2.3 d'Oracle. Jasper parse les fichiers JSP afin de les compiler en code Java en tant que servlets (gérés par Catalina). Pendant son exécution, Jasper est capable de détecter et recompiler automatiquement les fichiers JSP modifiés.

3.3. État de développement



Concepts architecturaux de Tomcat.

Les membres de la fondation Apache et des volontaires indépendants développent et maintiennent Tomcat. Les utilisateurs ont accès au code source et aux binaires sous la *Apache Software License*. La première version de Tomcat était la 3.0. En 2005, la version 5.x est sortie, implémentant les servlets 2.4 et les JSP 2.0. Depuis sa version 4.x, Tomcat utilise le serveur de conteneur Catalina.

L'architecture du logiciel se compose ainsi:

- Un serveur (server), soit tomcat en cours d'exécution ;
- Des services, intermédiaires collectant différents canaux de transmissions vers un traitement ;
- Un moteur (*engine*), qui pour chaque service traite les requêtes des collecteurs et renvoie les réponses ;
- Des hôtes (*host*), qui relient une adresse réseau avec le serveur ;
- Des connecteurs (*connector*), qui interprètent un canal et protocole de communication réseau à disposition des clients. Le connecteur HTTP est le plus typique ;
- Des contextes (*context*), qui sont les applications web.
- Le contexte est le lieu privilégié pour situer un service informatique que l'on veut rendre sur un réseau. Les autres modules sont mis en œuvre par le logiciel Tomcat lui-même.

3.4. Principe de fonctionnement :

Tomcat est souvent utilisé en association avec un autre serveur web plus généraliste, en général Apache, mais ce peut être aussi IIS :

Le serveur web s'occupe des pages web traditionnelles (.html, .php par exemple)

Il délègue à Tomcat les pages relevant spécifiquement d'une application web Java (Servlet, JSP...).

On peut utiliser le module `mod_jk` pour réaliser la communication entre Apache et Tomcat. Techniquement, Apache communique alors avec Tomcat sur le port 8009 (via le protocole `ajp13`).

Pour l'administration à distance, Tomcat inclut des applications accessibles par HTTP.

3.5. Arborescence de répertoires :

L'installation par défaut de Tomcat comprend les répertoires suivants :

bin : scripts et exécutables pour différentes tâches comme le démarrage et l'arrêt, notamment le fichier *catalina.sh* qui permet le lancement et l'arrêt du serveur tomcat.

common : classes communes que Catalina et les applications Web utilisent.

conf : fichiers de configuration au format XML et les DTD que ces fichiers XML utilisent.

lib : le répertoire contenant les bibliothèques logicielles (fichiers `.jar`).

logs : journaux des applications Web et de Catalina, notamment le fichier *catalina.out*.

server : classes utilisées seulement par Catalina.

shared : classes partagées par toutes les applications Web.

webapps : répertoire contenant les applications web (fichiers `.war`).

work : fichiers et répertoires temporaires (le cache).

3.6. Historique :

Le projet Tomcat a été lancé comme implémentation de référence des servlets par James Duncan Davidson, architecte logiciel chez Sun. Il a contribué à rendre le projet libre et a joué un rôle majeur dans sa donation par Sun à la fondation Apache.

Davidson aspirait dès le départ à rendre le projet libre. Comme la plupart des projets libres sont associés à un livre O'Reilly avec un animal en couverture, il a souhaité donner un nom d'animal au projet. Il a choisi le nom Tomcat (« matou ») car cet animal représentait quelque chose qui peut prendre soin de lui-même. Son souhait de voir une couverture d'animal s'est finalement concrétisé lorsqu'O'Reilly a publié un livre sur Tomcat avec un félin en couverture.

Bien qu'étant une implémentation de JEE moins complète que la plupart de ses concurrents, et même grâce à cette légèreté, Tomcat domine rapidement le marché des serveurs Java¹⁵. Il représentait en 2014 la moitié des utilisations de tels serveurs, loin devant JBoss et Jetty¹⁶. En 2016, son utilisation baisse à 42%.

3.7. Fonctionnalités des versions :

Tomcat 3.x (version initiale)

- Implémente les spécifications Servlet 2.2 et JSP 1.1.
- Rechargement des servlets.
- Fonctionnalités HTTP de base.

Tomcat 4.x

- Implémente les spécifications Servlet 2.3 et JSP 1.2.
- Nouveau conteneur de servlets Catalina.
- Nouveau moteur JSP Jasper.
- Connecteur Coyote.
- Java Management Extensions (JMX), JSP et administration Struts.

Tomcat 5.x

- Implémente les spécifications Servlet 2.4 et JSP 2.0.
- Ramasse-miettes réduit, performances et extensibilité améliorées.
- Wrappers natifs Windows et Unix.
- Analyse JSP plus rapide.

Tomcat 6.x

- Implémente les spécifications Servlet 2.5 et JSP 2.1.
- Support de Java 5.
- Amélioration de l'utilisation mémoire.

Tomcat 7.x

- Implémente les spécifications Servlet 3.0, JSP 2.2 et EL 2.2.
- Support de Java 6.
- Amélioration de la détection et de la prévention des fuites de mémoire.
- Mode hébergé simplifié.

Tomcat 8.x

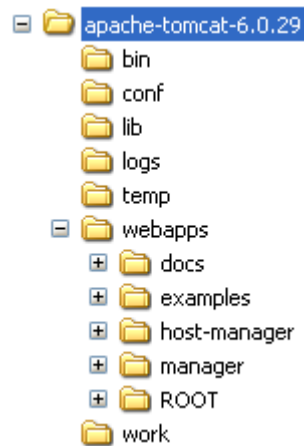
- Implémente les spécifications Servlet 3.1, JSP 2.3, EL 3.0 et WebSocket 1.0.
- Support de Java 7.
- Amélioration des performances: la version par défaut des connecteurs est désormais basée sur des entrées/sorties Java non bloquantes (Asynchronous I/O) à la place des versions bloquantes (Blocking I/O).
- Amélioration de la sécurité.

Tomcat 9.x

- Sont annoncés par la fondation Apache.
- Implémente Servlet 4.0 et WebSocket 1.1.
- Support de Java SE 8 et +.
- Améliorations de la stabilité de nombreuses API (javax, EL 3.0, Jasper 2, JSP 2.3...).

3.8. Organisation des répertoires de Tomcat :

L'installation de Tomcat est la même que l'on soit sous Windows ou Linux. Il suffit de télécharger le fichier archive (au format zip ou tar.gz) et de l'ouvrir dans un répertoire.



Organisation des répertoires de Tomcat

La distribution comporte donc sept répertoires, que nous allons passer en revue.

3.8.1. Répertoire bin

Le répertoire `bin` contient tous les fichiers qui permettent de lancer Tomcat, que ce soit sous Linux (fichiers `.sh`) ou Windows (fichiers `.bat`).

Notons l'utilitaire `version`, qui permet de tester la valeur des variables d'environnement "vues" par Tomcat, sa version, ainsi que différents détails sur l'environnement d'exécution.

3.8.2. Répertoire conf

Ce répertoire contient les fichiers de configuration de Tomcat. Nous reverrons la signification des fichiers qui se trouvent ici en détails. Les fichiers de ces répertoires permettent de configurer tous les aspects du fonctionnement de Tomcat, de sa sécurité, et aussi du chargement des applications web.

3.8.3. Répertoire lib

Ce répertoire contient les bibliothèques Java dont Tomcat a besoin pour fonctionner.

3.8.4. Répertoire log

Ce répertoire est vide. C'est ici que Tomcat écrit ses fichiers de journalisation dans sa configuration par défaut.

3.8.5. Répertoire temp

Répertoire temporaire contenant des fichiers temporaires, comme son nom le laisse supposer...

3.8.6. Répertoire webapp

Le répertoire `webapp` contient les applications web gérées par Tomcat. Cet endroit peut bien sûr être redéfini. Par défaut, il contient cinq applications, très utiles lorsque l'on débute :

docs : contient les pages de documentation de Tomcat, également accessibles en ligne ;

exemples : contient des exemples simples de servlets et de pages JSP ;

host-manager et manager : contiennent l'application de gestion des applications web de Tomcat. Cette application permet de charger des applications web à chaud, et de les visualiser dans une interface web.

ROOT : racine des applications web chargées par défaut.

3.8.7. Répertoire work

Répertoire de travail de Tomcat, dans lequel, entre autres, les classes Java correspondant aux pages JSP sont créées et compilées.

4. ACTIVE DIRECTORY

4.1. Caractéristiques d'Active Directory

Active Directory permet de recenser toutes les informations concernant le réseau, que ce soient les utilisateurs, les machines ou les applications. Active Directory constitue ainsi le moyen central de toute l'architecture réseau et a vocation à permettre à un utilisateur de retrouver et d'accéder à n'importe quelle ressource identifiée par ce service.

Active Directory est donc un outil destiné aux utilisateurs mais dans la mesure où il permet une représentation globale de l'ensemble des ressources et des droits associés il constitue également un outil d'administration et de gestion du réseau. Il fournit à ce titre des outils permettant de gérer la répartition de l'annuaire sur le réseau, sa duplication, la sécurisation et le partitionnement de l'annuaire de l'entreprise.

La structure d'Active Directory lui permet de gérer de façon centralisée des réseaux pouvant aller de quelques ordinateurs à des réseaux d'entreprises répartis sur de multiples sites.

4.2. Principe de fonctionnement d'Active Directory :

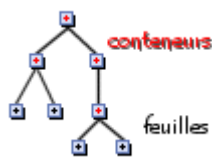
Active Directory permet de représenter et de stocker les éléments constitutifs du réseau (les ressources informatiques mais également les utilisateurs) sous formes d'objets, c'est-à-dire un ensemble d'attributs représentant un élément concret. Les objets sont organisés hiérarchiquement selon un **schéma** (lui-même stocké dans l'annuaire) définissant les attributs et l'organisation des objets.

Le service d'annuaire Active Directory permet de mettre ces informations à disposition des utilisateurs, des administrateurs et des applications selon les droits d'accès qui leur sont accordés.

4.3. Structure Active Directory :

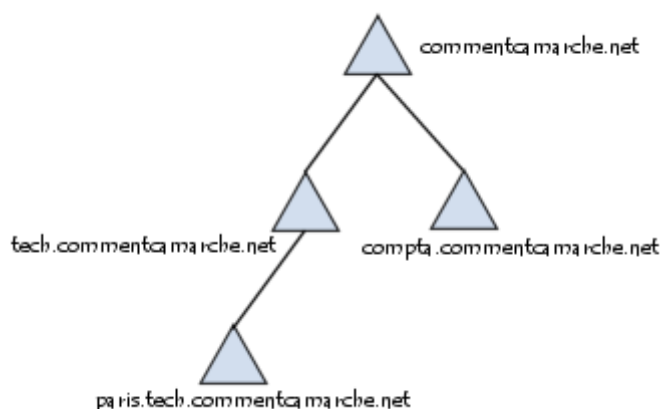
Les objets d'Active Directory (Utilisateurs, Groupes, Ordinateurs, etc.) correspondent à des classes, c'est-à-dire des catégories d'objets possédant les mêmes attributs. Ainsi un objet est une « instanciation » d'une classe d'objet, c'est-à-dire un ensemble d'attributs avec des valeurs particulières.

Lorsqu'un objet contient d'autres objets, on le qualifie de **conteneur**. Les conteneurs permettent de regrouper les objets dans une optique d'organisation. A l'inverse si l'objet est au plus bas niveau de la hiérarchie, il est qualifié de **feuille**.



La hiérarchie composée de l'ensemble des conteneurs (nœuds) et des feuilles est appelée **arbre**.

La notion d'arbre est étroitement liée à la notion de domaine, permettant de circonscrire des ressources informatiques dans un même périmètre de sécurité. Un domaine est ainsi constitué d'un ensemble défini d'éléments et possède une politique de sécurité (contrôles d'accès) qui lui est propre. Deux domaines (ou plus) possédant le même schéma peuvent établir entre eux des **relations d'approbation** (relations de confiance) bidirectionnelles et transitives basées sur le protocole Kerberos. L'ensemble des domaines reliés entre eux hiérarchiquement par des relations d'approbation constituent un **arbre de domaines** (appelé *arbre*). Le domaine situé au sommet de la hiérarchie est appelé « domaine racine » et les domaines situés en dessous sont des sous-domaines. Les domaines d'un même arbre partagent nécessairement le même espace de nom. Ainsi les domaines *commentcamarche.net*, *tech.commentcamarche.net*, *compta.commentcamarche.net* et *support.commentcamarche.net* font partie du même espace de nom et constituent un arbre lorsqu'ils sont liés par des relations d'approbation.



Le domaine *commentcamarche.net* représente ici le domaine racine.

On appelle « forêt » le regroupement (par relations d'approbation) de plusieurs arbres possédant le même schéma mais ne possédant pas nécessairement le même espace de nom, afin par exemple de joindre les annuaires de deux entreprises.

4.4. Les avantages des services de domaine Active Directory :

Pour l'administration de base de vos utilisateurs et ordinateurs réseau, l'utilisation d'AD DS présente plusieurs avantages.

Vous pouvez personnaliser la façon dont vos données sont organisées de façon à répondre aux besoins de votre entreprise. Si cela s'avère nécessaire, vous pouvez gérer AD DS à partir de n'importe quel ordinateur du réseau.

AD DS fournit une fonction intégrée de réplication et de redondance : si un contrôleur de domaine tombe en panne, un autre contrôleur de domaine prend la charge à son compte.

Tout accès aux ressources réseau passe par AD DS, ce qui assure une gestion centralisée des droits d'accès au réseau.

4.5. Services de domaine Active Directory (AD DS) : les termes à connaître :

Pour comprendre AD DS, il faut définir quelques termes clés.

Schéma : l'ensemble de règles utilisateur configurées qui régissent les objets et attributs dans AD DS.

Catalogue global : le conteneur de tous les objets AD DS. Si vous avez besoin de trouver le nom d'un utilisateur, ce nom est stocké dans le catalogue global.

Mécanisme de requête et d'index : ce système permet aux utilisateurs de se trouver les uns les autres dans AD. Par exemple, lorsque vous commencez à saisir un nom dans votre client de messagerie, ce dernier vous propose les correspondances possibles.

Service de réplication : le service de réplication garantit que tous les contrôleurs de domaine du réseau partagent le même catalogue global et le même schéma

Sites : les sites sont des représentations de la topologie réseau ; AD DS sait ainsi quels sont les objets qui vont ensemble, ce qui lui permet d'optimiser la réplication et l'indexation.

Lightweight Directory Access Protocol : LDAP est un protocole qui permet à AD de communiquer avec d'autres services d'annuaire sur d'autres plates-formes.

4.6. Quels sont les services fournis par AD DS ?

Voici les services fournis par AD DS, qui constituent les fonctionnalités de base d'un système de gestion centralisée des utilisateurs.

Services de domaines : stocke les données et gère les communications entre les utilisateurs et le contrôleur de domaine. Il s'agit de la principale fonctionnalité d'AD DS.

Services de certificat : permet à votre contrôleur de domaine de servir des certificats et des signatures numériques, ainsi qu'un chiffrement à clé publique.

Lightweight Directory Services : prend en charge LDAP pour des services de domaine multiplateformes, par exemple l'ensemble des ordinateurs Linux présents sur votre réseau.

Services de fédération d'annuaire : dans la même session, fournit une authentification SSO pour plusieurs applications. Ainsi, les utilisateurs ne sont pas obligés de ressaisir les mêmes identifiants.

Gestion des droits : contrôle les politiques en matière de droits à l'information et d'accès aux données. Par exemple, la gestion des droits détermine si vous pouvez accéder à un dossier ou envoyer un e-mail.

5. MARIADB

5.1. Présentation:

MariaDB est un système de gestion de base de données édité sous licence GPL. Il s'agit d'un fork communautaire de MySQL qui fait suite au rachat de MySQL par Sun Microsystems en 2009 et des annonces du rachat de Sun Microsystems par Oracle Corporation. La gouvernance du projet est assurée par la fondation MariaDB, et sa maintenance par la société Monty Program AB, créateur du projet. Cette gouvernance confère au logiciel l'assurance de rester libre.

Depuis Fedora 19, MySQL a été complètement remplacé par MariaDB. Cependant dans une volonté de complète compatibilité les commandes sous appellation mysql ont été conservées.

L'utilisation de phpMyAdmin est donc compatible, vous pouvez consulter le tutoriel phpMyAdmin.

MariaDB est une version améliorée et pouvant être utilisée en remplacement direct de MySQL est disponible sous la license GPL v2.

5.2. Historique :

En 2009, à la suite du rachat de MySQL par Sun Microsystems et des annonces du rachat de Sun Microsystems par Oracle Corporation, Michael Widenius, fondateur de MySQL, quitte cette société²² pour lancer le projet MariaDB, dans une démarche visant à remplacer MySQL tout en assurant l'interopérabilité. Le nom vient de la 2e fille de Michael Widenius, Maria (la première s'appelant My)²³.

L'encyclopédie libre Wikipédia annonce, fin 2012, la migration de MySQL à MariaDB²⁴. Les distributions GNU/Linux, comme Fedora, openSuse et Debian abandonnent elles aussi MySQL. En septembre 2013, Google annonce l'adoption de MariaDB en lieu et place de MySQL. Dans la foulée un des ingénieurs de Google est affecté à la Fondation MariaDB²⁵.

Un consortium baptisé Open Database Alliance a par ailleurs été créé pour assurer le développement du logiciel (comparable à la fondation Linux avec le noyau Linux).

Runa Capital a investi dans l'entreprise en 2015²⁶.

5.3. Sécurité :

La sécurité est l'un des principaux objectifs des développeurs de MariaDB. Le projet maintient ses propres jeux de patches de sécurité en plus de ceux de MySQL. Pour chaque version de

MariaDB, les développeurs incluent également les patches de sécurité de MySQL et les améliorent si nécessaire. Quand des failles de sécurité critiques sont découvertes, les développeurs préparent et distribuent immédiatement une nouvelle version de MariaDB afin de régler le problème aussi vite que possible.

Un grand nombre de failles de sécurité découvertes dans MySQL et MariaDB ont été découvertes et reportées par l'équipe de MariaDB. L'équipe de MariaDB travaille en proche collaboration avec <http://cve.mitre.org/> afin de s'assurer que toutes les failles de sécurité sont reportées rapidement et avec suffisamment de détails. Les détails sur les failles sont habituellement divulgués après que les versions corrigées de MariaDB ainsi que de MySQL ait été publiées.

5.4. Compatibilité :

MariaDB est maintenu actualisé en permanence à partir de la dernière version de MySQL de la même branche et à bien des égards MariaDB fonctionnera exactement comme MySQL. Toutes les commandes, les interfaces, les bibliothèques et les APIs qui existent dans MySQL existent également dans MariaDB. Il n'y a pas besoin de convertir les bases de données pour passer à MariaDB. MariaDB peut être utilisé en remplacement direct de MySQL ! En outre, MariaDB dispose de beaucoup de nouvelles fonctionnalités dont vous pouvez tirer profit.

5.5. Versions actuelles & Calendrier des sorties :

- La version stable actuelle de MariaDB est MariaDB 10.0.
- La version en développement est MariaDB 10.1.
- La précédente version stable est MariaDB 5.5.

Les versions stables et de développement de MariaDB (ainsi que de nombreuses versions précédentes) peuvent être téléchargées sur <http://downloads.mariadb.org/>.

Nous faisons des mises à jour aux versions actives tous les mois environ. Les sorties majeures sont faites tous les 9 mois environ. Les versions 10.x sortiront probablement tous les 6 mois.

La roadmap sur <http://mariadb.org/jira/> vous permettra de savoir quand est prévue la prochaine version de MariaDB. Cela n'est évidemment qu'une estimation que les développeurs tenteront de suivre dans la mesure du possible.

5.6. Exécution et gestion des licences MariaDB :

Le logiciel MariaDB fonctionne sous Windows, Linux et macOS ; il est disponible sur les plateformes matérielles Intel et IBM Power8, et s'exécute en tant que service sur plusieurs plateformes cloud. Les langages de programmation pris en charge comprennent C++, C#, Java, Python et d'autres encore.

Une implémentation MariaDB Galera Cluster sous Linux a également été développée pour fournir une solution de cluster multimaître synchrone aux utilisateurs de MariaDB. Toutefois, l'API qui connecte la base de données à Galera Cluster, une autre technologie open source, est incluse par défaut dans MariaDB à partir de la version 10.1, ce qui évite de télécharger séparément les clusters.

MariaDB est fournie en tant que logiciel open source au titre de la version 2 de la licence GNU General Public License (GPL), tout comme le moteur MariaDB ColumnStore, qui est conçu pour une utilisation dans les applications de Big Data. MariaDB Corp. propose également une fonctionnalité de proxy de base de données appelée MaxScale, qui permet de fractionner les requêtes sur plusieurs serveurs MariaDB ; cette technologie est disponible sous une licence BSL (Business Source License) créée par la société, qui consiste à facturer MaxScale dans les déploiements comportant plus de trois serveurs. Cependant, les versions du logiciel devraient passer en open source via la licence GPL dans les quatre ans après leur sortie.

Comme d'autres SGBDR open source, tels PostgreSQL et Firebird, MariaDB et MySQL se sont imposées comme substituts à bas coût aux bases de données majoritaires Oracle, Microsoft SQL Server et IBM DB2. Les bases de données open source sont largement utilisées dans les applications web et cloud ; MariaDB, en particulier, a fait des adeptes parmi les utilisateurs d'autres composants de logiciels open source, notamment dans l'infrastructure OpenStack. Les utilisateurs de MariaDB comptent dans leurs rangs Wikipedia, Facebook et Google, ainsi que la DBS Bank à Singapour et l'Institute for Health Metrics and Evaluation à Seattle.

6. MYSQL

6.1. Définition :

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde³, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, PostgreSQL et Microsoft SQL Server.

Son nom vient du prénom de la fille du cocréateur Michael Widenius, My (**sv**) (prononcer [my]). SQL fait référence au *Structured Query Language*, le langage de requête utilisé.

MySQL AB a été acheté le 16 janvier 2008 par Sun Microsystems pour un milliard de dollars américains⁴. En 2009, Sun Microsystems a été acquis par Oracle Corporation, mettant entre les mains d'une même société les deux produits concurrents que sont Oracle Database et MySQL. Ce rachat a été autorisé par la Commission européenne le 21 janvier 2010^{5,6}.

Depuis mai 2009, son créateur Michael Widenius a créé MariaDB (Maria est le prénom de sa deuxième fille) pour continuer son développement en tant que projet Open Source.

6.2. Historique :

David Axmark est le fondateur de MySQL. La première version de MySQL est apparue le 23 mai 1995. Il a d'abord été créé pour un usage personnel à partir de mSQL en s'appuyant sur le langage de bas niveau ISAM qu'ils trouvaient trop lent et trop rigide. Ils ont créé une nouvelle interface SQL en gardant la même API que mSQL.

MySQL est passé en licence GPL à partir de la version 3.23.19 (juin 2000)

Version 4.0 : première version en octobre 2001, stable depuis mars 2003

Version 4.1 : première version en avril 2003, stable depuis octobre 2004

Version 5.0 : première version en décembre 2003, stable depuis octobre 2005

Version 5.1 : première version en novembre 2005, Release Candidate distribuée depuis septembre 2007

Version 5.2 : distribuée en avant-première (ajout du nouveau moteur de stockage *Falcon*) en février 2007, cette ligne a ensuite été renommée 6.0

Version 5.5 : Version stable depuis octobre 2010

Version 5.6 : Version stable depuis février 2013

Version 5.7 : Version stable depuis octobre 2015¹⁸

Version 6.0 : première version alpha en avril 2007, abandonnée depuis le rachat de MySQL par Oracle en décembre 2010

Version 8.0 : Version stable depuis avril 2018¹⁹

En 2003, MySQL AB et SAP AG concluent un accord. Le résultat s'appellera MaxDB. Il est issu de l'intégration du système de SAP (SAP DB) dans MySQL.

Le 20 avril 2009, Oracle Corporation annonce racheter Sun Microsystems pour 7,4 milliards de dollars²⁰, créant la crainte de voir MySQL disparaître – ou du moins ne plus être développé – au profit d'Oracle, le SGBD d'Oracle Corporation. MySQL est cependant loin d'avoir toutes les fonctionnalités d'Oracle, et pourrait donc être vu au contraire par la société comme un produit gratuit d'appel banalisant l'usage de SQL et préparant donc ses futures ventes.

6.3. Dénomination :

MySQL est l'œuvre d'une société suédoise, MySQL AB, fondée par David Axmark, Allan Larsson et Michael « Monty » Widenius.

Le nom MySQL vient de la combinaison de *My*, fille du cofondateur Michael « Monty » Widenius, avec l'acronyme SQL (selon la documentation de la dernière version en anglais¹⁶). D'autre part, le dossier base et une grande partie des bibliothèques utilisées par les développeurs utilisent le préfixe `My`.

Le nom du logo de MySQL (le dauphin) Sakila, a été choisi par les créateurs de MySQL sur la base d'un concours. Il a été proposé par Ambrose Twebaze, développeur du Swaziland. D'après Ambrose, le nom Sakila puise ses origines du siswati, la langue locale du Swaziland.

6.4. Utilisation :

MySQL fait partie du quatuor LAMP : Linux, Apache, MySQL, PHP. Il appartient également à ses variantes WAMP (Windows) et MAMP (Mac OS).

Le couple PHP/MySQL est très utilisé par les sites web et proposé par la majorité des hébergeurs Web. Plus de la moitié des sites Web fonctionnent sous Apache, qui est le plus souvent utilisé conjointement avec PHP et MySQL.

6.5. Caractéristiques :

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateur.

C'est un logiciel libre⁷, open source⁸, développé sous double licence selon qu'il est distribué avec un produit libre ou avec un produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Un logiciel qui intègre du code MySQL ou intègre MySQL lors de son installation devra donc être libre ou acquérir une licence payante. Cependant, si la base de données est séparée du logiciel propriétaire qui ne fait qu'utiliser des API tierces (par exemple en C# ou php), alors il n'y a pas besoin d'acquérir une licence payante MySQL. Ce type de licence double est utilisé par d'autres produits comme le framework de développement de logiciels Qt (pour les versions antérieures à la 4.5).

6.5.1. Systèmes d'exploitation supportés :

MySQL fonctionne sur de nombreux systèmes d'exploitation différents, incluant AIX, IBM i-5, BSDi, FreeBSD, HP-UX, Linux, Mac OS X, NetWare, NetBSD, OpenBSD, OS/2 Warp, SGI IRIX, Solaris, SCO OpenServer, SCO UnixWare, Tru64 Unix, Windows.

Les bases de données sont accessibles en utilisant les langages de programmation C, C++, VB, VB.NET, C#, Delphi/Kylix, Eiffel, Java, Perl, PHP, Python, Windev, Ruby et Tcl ; une API spécifique est disponible pour chacun d'entre eux. Une interface ODBC appelée MyODBC est aussi disponible. En Java, MySQL peut être utilisé de façon transparente avec le standard JDO.

6.5.2. Langages informatiques supportés :

MySQL supporte deux langages informatiques, le langage de requête SQL et le SQL/PSM (Persistent Stored Modules), une extension procédurale standardisée au SQL incluse dans la norme SQL:20039. SQL/PSM, introduit dans la version 5 de MySQL, permet de combiner des requêtes SQL et des instructions procédurales (boucles, conditions...), dans le but de créer des traitements complexes destinés à être stockés sur le serveur de base de données (objets serveur), par exemple des procédures stockées ou des déclencheurs.

6. Version de SQL :

MySQL supporte la norme SQL212 (utilisation des RIGHT JOIN et LEFT JOIN¹³), la conformité à cette norme garantissant qu'il honorera les requêtes normalisées correspondantes. Cependant, les fonctionnalités des normes SQL les plus récentes ne sont pas toutes implémentées¹⁴ et certaines ne respectent pas la syntaxe recommandée¹⁵ (la concaténation par exemple), empêchant l'interopérabilité des requêtes entre différents SGBD.

7. Fonctionnalités :

Deux moteurs principaux sont présents dans MySQL : MyISAM et InnoDB.

MyISAM, contrairement à InnoDB, ne supporte ni transactions ni intégrité automatique des tables, il n'est pas destiné aux applications dont la cohérence des données est critique ; cependant, ses performances le font adopter pour des applications ayant besoin d'une base de données simple et peu onéreuse à mettre en œuvre.

Pour les utilisateurs, phpMyAdmin est un outil web souvent disponible pour créer, remplir et utiliser des bases MySQL.

8. Interopérabilité:

Les tables des bases de données SQL et celles des tableurs communiquent bien dans le deux sens (en ce qui concerne les seules **données** numériques et chaînes de caractères) au moyen du format CSV. Des formats de sauvegarde et restauration existent, sous forme d'ordres SQL, ainsi qu'en format XML.

9. Usage en production:

L'intérêt d'utiliser des outils ouverts a également poussé quelques établissements bancaires tels que le Crédit mutuel, le Crédit agricole, le Crédit lyonnais, la Banque de développement du Canada à inscrire MySQL au catalogue de leurs produits informatiques internes.

Conclusion générale

A la fin de notre projet on réalise à quel point un bon système de supervision peut fournir une aide précieuse à l'administrateur dans le contrôle de son réseau. Notre travail nous a permis de définir l'objectif de la supervision et son influence sur le système informatique et sur le fonctionnement performant des entreprises.

Les grandes entreprises nécessitent beaucoup de matériel. Ce dernier doit être géré par l'administrateur, ce qui est difficile s'il ne connaît pas tous les détails sur son parc informatique en temps réel. C'est pourquoi un bon logiciel de supervision facilite le travail de l'administrateur et le réduit à des simples vérifications ou des actions de correction pour les problèmes.

La surveillance en tout moment évitera à l'entreprise les erreurs et les pannes qui causent des interruptions au niveau de ces fonctionnalités et qui ont un impact préjudiciable sur son activité et sa réputation. Nous avons donc donné toutes les étapes nécessaires de l'installation et de configuration d'un service de supervision Zabbix.

Zabbix est un logiciel libre et complet qui fonctionne sous Linux et qui réalise le principe de la supervision.

Il utilise des modèles prédéfinis qui peuvent être modifiés par l'administrateur pour communiquer avec les machines (hôtes) et il schématise les composantes du réseau. Il est très facile à manipuler. Les différents rapports sur l'état des machines supervisées sont stockés dans une base de données.

De plus notre projet peut être développé par ceux qui veulent continuer ce travail pour rendre le système plus performant.

Webographie

<https://wiki.monitoring-fr.org/zabbix/zabbix-introduction>

<https://www.zabbix.com/documentation/4.0/fr/manual/introduction/about>

https://helenemoutinho.files.wordpress.com/2017/06/projet_zabbix.pdf

<https://www.zabbix.com/documentation/4.0/fr/manual/introduction/features>

<https://fr.wikipedia.org/wiki/Node.js>

https://fr.wikipedia.org/wiki/Apache_Tomcat

<http://blog.paumard.org/cours/servlet/chap02-presentation-tomcat-installation.html>

<https://web.maths.unsw.edu.au/~lafaye/CCM/activedirectory/active-directory-intro.htm>

<https://web.maths.unsw.edu.au/~lafaye/CCM/activedirectory/active-directory-principes.htm>

<https://blog.varonis.fr/services-de-domaine-active-directory-ad-ds-presentation-et-fonctions/>

https://doc.fedora-fr.org/wiki/Installation_et_configuration_de_MariaDB#Pr.C3.A9sentation_de_MariaDB

<https://fr.wikipedia.org/wiki/MariaDB>

<https://mariadb.com/kb/fr/a-propos-de-mariadb/>

<https://fr.wikipedia.org/wiki/MySQL>