**CONTEXTE TECHNOLOGIQUE :**

L'approche DevOps et la surveillance distribuée des applications basées sur l'architecture à microservices sont des concepts liés à la gestion des infrastructures informatiques et au développement logiciel.

**DevOps** : méthodologie qui vise à intégrer les équipes de développement et d'exploitation (opérations) dans un processus collaboratif continu. L'objectif est d'automatiser et d'accélérer le développement, les tests, le déploiement et la maintenance des applications logicielles. Cela permet d'améliorer l'efficacité et la qualité des livraisons logicielles, ainsi que la stabilité et la disponibilité des systèmes.

**L'architecture à microservices** : c’est un style d'architecture logicielle dans lequel une application est décomposée en un ensemble de services indépendants et autonomes, qui peuvent être développés, déployés et mis à l'échelle de manière indépendante. Chaque service est responsable d'une fonction spécifique et communique avec les autres services via des interfaces bien définies.

**Surveillance distribuée :**

Elle est essentielle pour assurer le bon fonctionnement et la performance des systèmes distribués complexes. Étant donné que chaque microservice fonctionne de manière autonome et peut être développé et déployé indépendamment, il devient crucial de surveiller et de collecter des métriques de performance et de disponibilité à partir de chaque service.

**PROBLEMATIQUE SOCIETALE :**

La problématique sociétale liée à l'approche DevOps et à la surveillance distribuée des applications basées sur les architectures à microservices concerne principalement la stabilité et la fiabilité des systèmes informatiques utilisés dans divers domaines, tels que les services bancaires en ligne, le commerce électronique, les soins de santé, les transports, etc.

**LES ENJEUX :**

Les enjeux associés à cette problématique sont les suivants :

1. Stabilité et disponibilité : Les architectures à microservices permettent une meilleure résilience et une haute disponibilité des applications, mais elles introduisent également des défis de gestion de la complexité. Les enjeux consistent à surveiller en temps réel chaque microservice pour détecter les défaillances et les problèmes de performance, afin de garantir une expérience utilisateur continue.
2. Évolutivité : Les architectures à microservices offrent une flexibilité pour développer et déployer indépendamment chaque service. Cependant, cela rend également nécessaire la mise en place de mécanismes de surveillance et de gestion de la scalabilité afin de maintenir les performances du système tout en ajoutant ou supprimant des services.

**Références :**

**[ 1 ]** "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", Gene Kim, Jez Humble, Patrick Debois et John Willis.

**[ 2 ]** “Design of Modern Distributed Systems based on Microservices Architecture”, Isak Shabani, Endrit Mëziu, Blend Berisha, Tonit Biba.

**[ 3 ] “**Monitoring tools for DevOps and microservices: A systematic grey literature review”, L. Giamattei a, A. Guerriero et al