Pitch Sécurisation Docker

in good we trust.

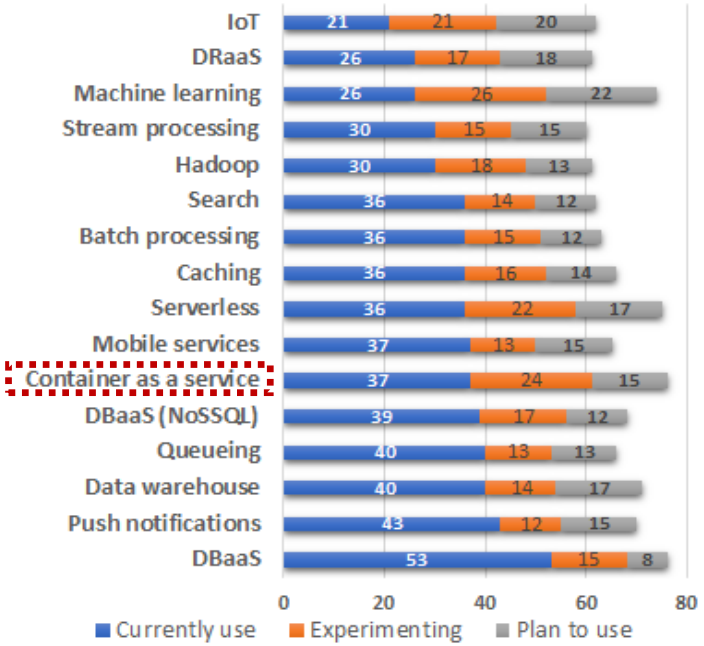
Sécurité Docker

Contexte

- Application Cloud native
- CI/CD
- DevSecOps
- Agile
- Scalabilité
- Micro-services

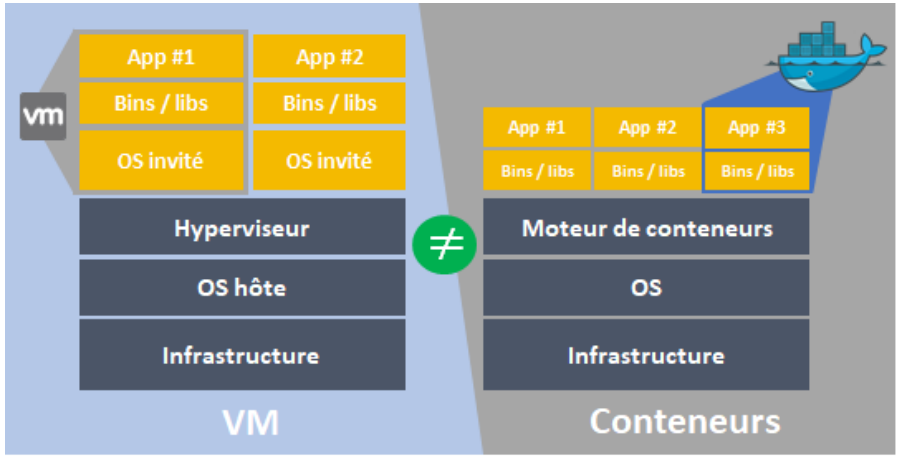
Utilisation des services Cloud

Container as a Service identifié par les DSI parmi les priorités en terme d'utilisation / expérimentation / prévision des XaaS



Source: RightScale 2019 State of the Cloud Report from Flexera

VM vs Conteneur



Sécurité Docker

Enjeux et Bénéfices

Les conteneurs répondent au pic de charge par **la scalabilité horizontale**

Etant plus léger, les conteneurs permettent des **déploiement plus rapides d'applications (CI/CD)**

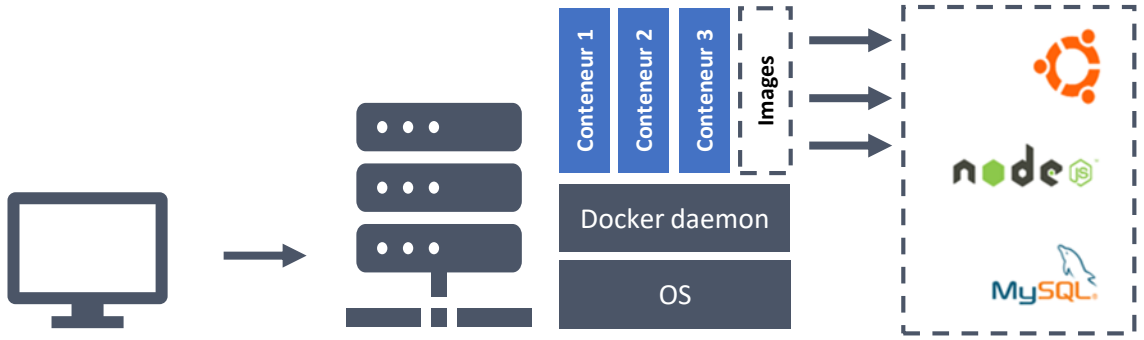
Les conteneurs favorisent les **architectures micro-services**, où les applications sont découpées en sous services simples

Les conteneurs offrent une **meilleure portabilité** (plus de différence entre environnements de prod et dev) avec **l'immutabilité du conteneur**

Les conteneurs offrent une **meilleure stabilité**, on ne met pas à jour un container, on le tue et on le remplace

Etant plus léger que la VM, les conteneurs permettent une **meilleure consolidation** des machines

Sécurité Docker: Composants



L'architecture Docker est composée principalement des éléments suivants

- **Les dockerfiles** : Fichiers de configuration qui décrivent exactement ce qui doit être installé dans le conteneur.
- **Les images** : Templates prêt à l'emploi (Java, une base de données, un script, etc…), mais est dans un état inerte.
- **Les conteneurs** : Templates instanciés et configurés, de façon unique ou multiple
- **Une registry Docker** : Application qui permet de stocker et distribuer des images Docker

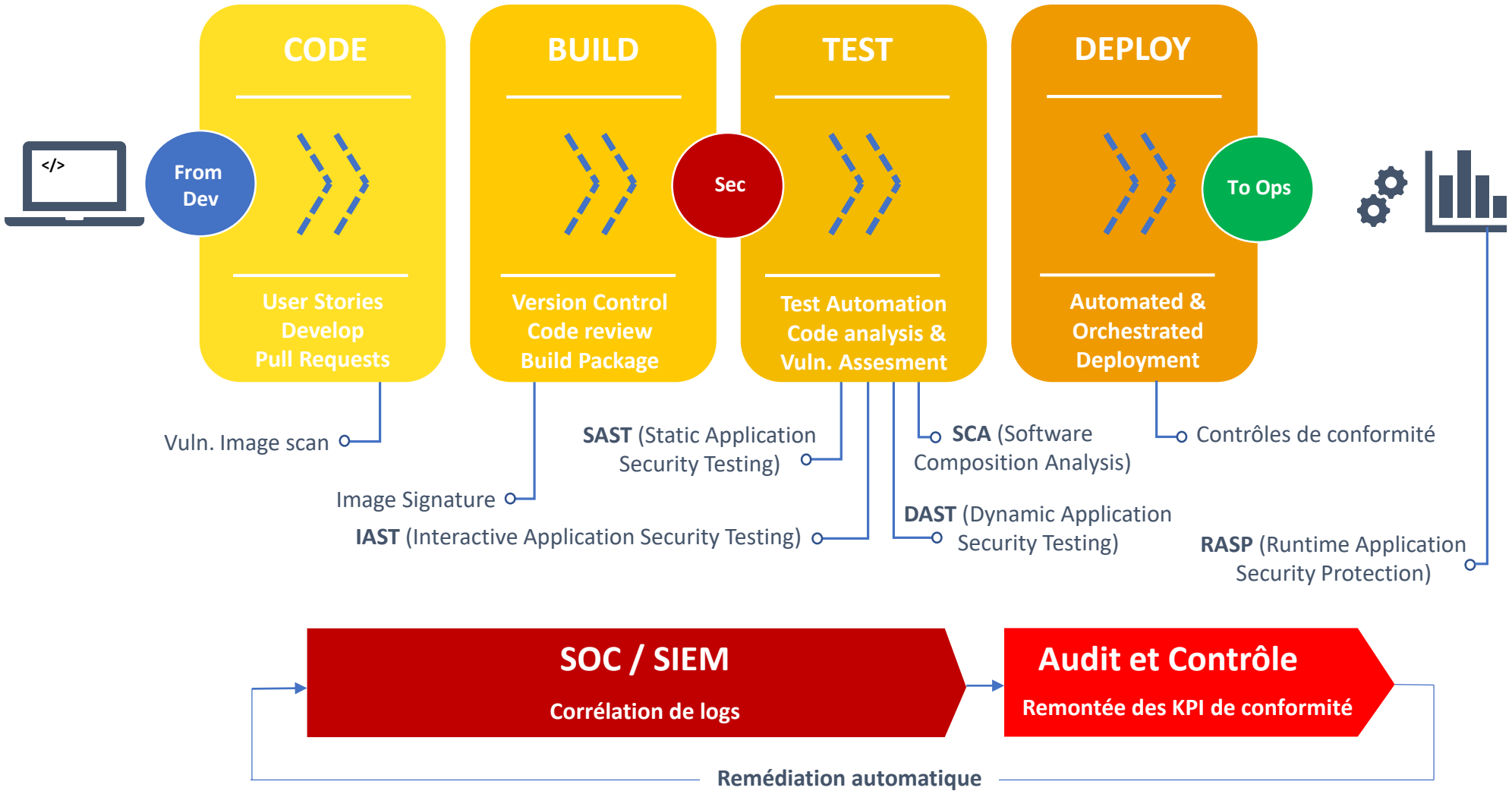
Complexité inhérente aux modèles des conteneurs

Afin de pouvoir **utiliser pleinement les fonctionnalités des conteneurs** (scalabilité automatique, déploiements sans interruption de service, self-healing) les conteneurs doivent être **supervisés par une plateforme d'orchestration.**

Sécurité Docker: Quelques exemples de menace

Vulnérabilités	Menaces	Impacts	Remédiations
Absence ou manque de durcissement de la machine hôte	Exploitation d'une vulnérabilité sur la machine hôte	Corruption / fuite de données, rebond vers un conteneur	Sécurisation de la machine hôte (restrictions des accès, authentification, chiffrement des flux) - SELinux
Défaut d'intégrité des images Docker	Exploitation d'une vulnérabilité d'une image Docker	Corruption / fuite de données	Validation des images Docker (vulnérabilités, signature d'image,…) – Notary
Défaut d'isolation du conteneur	Exfiltration de données	Corruption / fuite de données, rebond vers un conteneur	Restriction des accès, principe du moindre privilège – Namespace / Seccomp, Network policies
Absence de quota d'utilisation des ressources	Surexploitation des ressources de la plate-forme	Déni / interruption de service	Définition de quotas d'utilisation des ressources
Vulnérabilité des images Docker	Altération, exfiltration de données	Corruption, fuite de données	Analyse et gestion des vulnérabilités des images – CoreOS
Mauvaise Gestion des secrets	Altération / exfiltration des données / du paramétrage de la plate-forme	Fuite de données, modification de paramètres / des ressources du conteneur	Mise en place d'un coffre fort pour la gestion des secrets – Vault
Absence de supervision de la plate-forme	Exploitation de ressources, exfiltration de données	Détournement des fonctions usuelle / Déni de service, Fuite de données	Analyse dynamique des conteneurs, supervision de la plate-forme – SOC / SIEM, audit logs, Falco
Gestion des accès défaillante	Utilisation du compte administrateur local par une personne non autorisée	Modification des paramètres ou suppression de machines, suppression de machines, de données	Utiliser un annuaire centralisé pour gérer les authentifications et supprimer les comptes user locaux, principe du moindre privilège - Linux capabilities

Sécurité Docker: Cycle de vie des images



Sécurité Docker: Pour aller plus loin

Ansible-hardening utilise des guides de durcissement (STIG, DISA) de sécurité standard pour sécuriser les hôtes Linux.

Permet d'automatiser les tests de sécurité, en veillant à la conformité aux normes dans chaque environnement et à chaque étape du développement.

CoreOS, permet de tester la présence de failles et vulnérabilités de sécurité dans les images Docker.

Cloud Security Alliance (CSA) et le enter for Internet Security proposent et font la promotion des guides d'hygiène et de bonnes pratiques de mise en œuvre des technologies avancées et notamment sur les environnements Cloud.

Guide de sécurité Docker	Guide de sécurité Kubernetes
Official Docker Documentation (lien) CIS Docker hardening guide (lien) SANS Docker Security Checklist (lien) CSA Best Practice for Implementing a Secure Application Container Architecture (lien) CIS Docker Benchmark (lien) Docker Security – Using Container Safely in Production (lien) Docker release (lien)	Kubernetes Best Practices (lien) CIS Kubernetes Benchmark (lien) Kubernetes Reference Guide (lien) Kubernetes Cookbook (lien)