Vous placez vos applications dans des conteneurs, mais sont-ils protégés contre les menaces ?

Guide des possibilités DevOps.
Risques liés à la sécurité.

kaspersky

Contenu

1. Qu'est-ce qu'un conteneur ?

Un conteneur est une unité de logiciel autonome et exécutable qui regroupe tout ce qui est nécessaire à l'exécution d'une application (code, outils et bibliothèques du système, binaires et paramètres), ce qui permet à l'application de fonctionner rapidement, de manière transparente et uniforme dans n'importe quel environnement informatique, qu'il soit sur site ou dans le cloud.

Serveurs physiques (années 1990)

Architectures client-serveur sur des serveurs physiques utilisant un seul système d'exploitation et ne gérant souvent qu'une seule application. La croissance exponentielle des applications a mis en évidence les problèmes chroniques d'allocation des ressources des serveurs physiques.

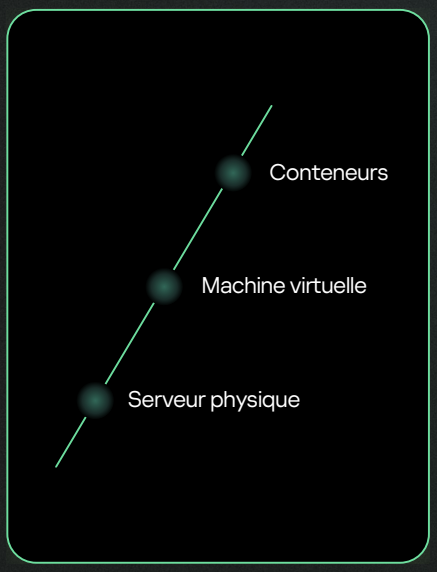
Machines virtuelles (années 2000)

Les logiciels de virtualisation ont rendu les environnements informatiques indépendants de l'infrastructure physique en faisant tourner plusieurs instances de systèmes d'exploitation sur un seul serveur, ce qui a permis aux ordinateurs de partager des ressources matérielles avec des environnements numériquement séparés.

Conteneurs (à partir de 2015)

La conteneurisation pousse la virtualisation encore plus loin en permettant aux applications de fonctionner dans des environnements dédiés avec le même système d'exploitation sur une seule machine virtuelle ou un seul serveur. Gartner prévoit que l'adoption de Kubernetes (orchestration de conteneurs) dépassera 75 %[1].

Terminologie clé :

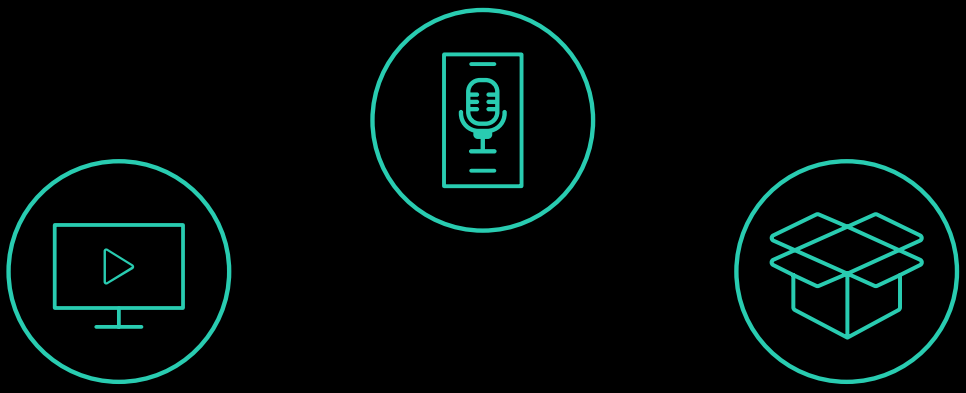


2. Cas d'usage quotidien de la technologie des conteneurs

Pratiquement tous les types d'applications peuvent être mis en conteneur, mais les applications fonctionnant à l'aide de micro-services s'y prêtent particulièrement. Chaque micro-service est développé de façon isolée, puis intégré à d'autres conteneurs de manière à construire une application complète.

Qu'il s'agisse de votre application musicale favorite, d'un service de livraison ou de solutions industrielles complexes, tous sont probablement construits à partir de conteneurs.

[1] https://medium.com/@IntelliSoft/what-is-kubernetes-and-when-to-use-it-key-trends-in-2023-aa8cd2eb5053

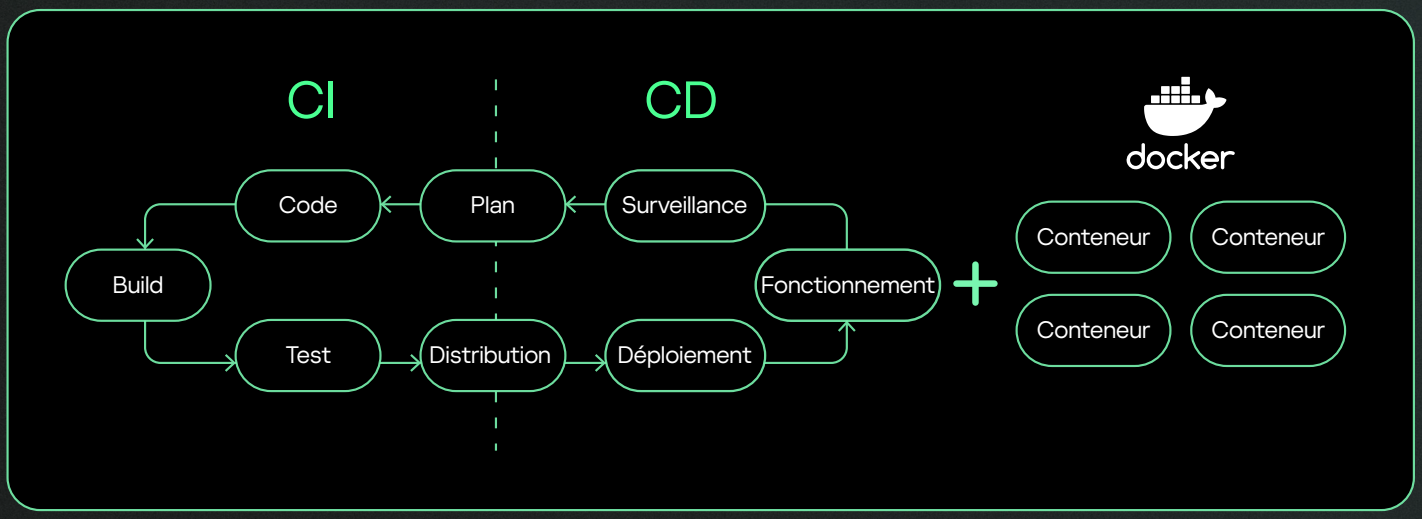


3. Cycle de vie du développement logiciel (CI/CD) et conteneurs

L'intégration et le déploiement continus (CI/CD) se sont rapidement imposés comme les pratiques de développement les plus courantes.

L'intégration continue (CI) est un ensemble de pratiques de codage automatisant la construction et le test d'applications permettant aux équipes de développeurs de concevoir des applications sans interruption, en modifiant le code de manière continue, ce qui permet d'améliorer la collaboration ainsi que la qualité du code.

Le déploiement continu (CD) permet d'automatiser les modifications de codage du développement, des tests et du déploiement des applications.



Où les conteneurs viennent compléter les CI/CD

Normalisé – capable de fonctionner partout.

Léger – partage le même noyau de système d'exploitation.

Sécurisé – logiciel isolé de l'environnement.

La conteneurisation (le fait de placer un module logiciel, son environnement, ses dépendances et sa configuration dans un conteneur) permet d'accélérer le processus de conception et de déploiement des applications, multipliant ainsi les avantages du cycle d'intégration et de déploiement continus du développement logiciel :

- accélère la rédaction, le débogage et le lancement d'une version.
- améliore la stabilité de l'application.
- réduit les exigences en matière d'infrastructure pour le développeur et le client.
- s'adapte sans effort à des projets de toutes tailles.
- orchestre le déploiement de conteneurs pour obtenir des grappes de conteneurs duplicables et continues.

4. Composants de l'infrastructure des conteneurs

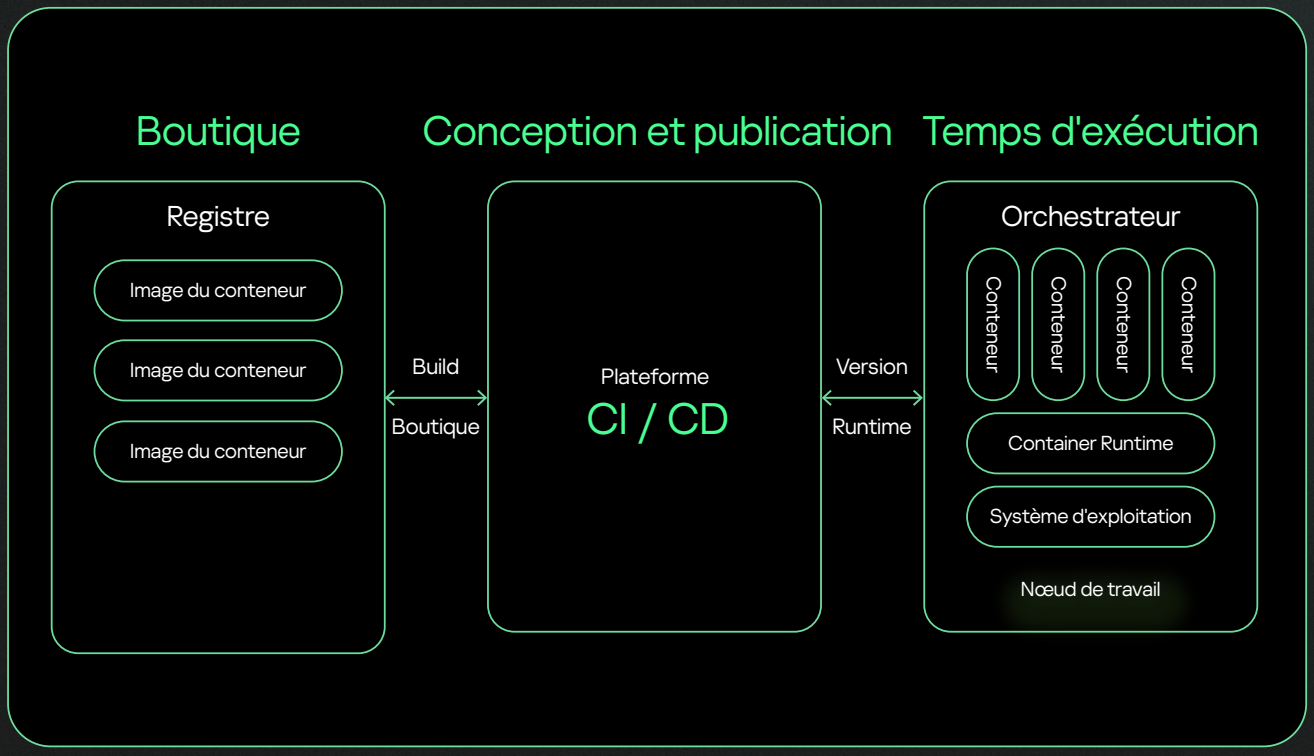
Image : Module central d'une architecture mise en conteneur, une image est un fichier statique et immuable contenant un code exécutable déployé de manière isolée dans n'importe quel environnement.

Registre d'images : Stockage permettant de conserver des images de conteneurs et d'y accéder. Les registres des conteneurs peuvent prendre en charge le développement d'applications basées sur des conteneurs, souvent dans le cadre de processus DevOps. Parmi les registres d'images, citons Docker Hub, JFrog, Sonatype Nexus OSS ou encore GitLab Registry. Les registres peuvent être publics ou privés.

Orchestrateur : Outil de déploiement et de gestion d'un grand nombre de conteneurs permettant aux développeurs d'utiliser des clusters de conteneurs pour passer à l'échelle et automatiser le processus de connexion des conteneurs en groupes. Parmi les exemples d'orchestrateurs, on peut citer Kubernetes, OpenShift et Docker Swarm.

Conteneur : Unité de logiciel autonome et exécutable qui regroupe tout ce qui est nécessaire à l'exécution d'une application (code, outils et bibliothèques du système, binaires et paramètres), ce qui permet à l'application de fonctionner rapidement, de manière transparente et uniforme dans n'importe quel environnement informatique, qu'il soit sur site ou dans le cloud.

Système d'exploitation : Programme hôte qui gère tous les autres programmes d'application d'un ordinateur.



5. Principaux risques liés à la sécurité des conteneurs

des personnes interrogées ont indiqué avoir constaté une perte de revenus ou de clientèle à la suite d'un incident lié à la sécurité des conteneurs.[2]

des entreprises ont été confrontées à au moins deux incidents dans Kubernetes au cours des 12 derniers mois.[3]

Images

- Sources externes ouvertes
- Vulnérabilités logicielles
- Erreurs de configuration
- Programme malveillant
- Secrets dans le domaine public
- Utilisation d'images non fiables

Registre d'images

- Connexions non sécurisées
- Présence d'images obsolètes contenant des vulnérabilités et des programmes malveillants
- Restrictions insuffisantes en matière d'authentification et d'autorisation

Orchestrateur

- Accès administratif illimité
- Accès sans autorisation
- Pas ou peu de séparation du trafic entre les conteneurs
- Les conteneurs présentant différents niveaux de protection des données ne sont pas espacés par les hôtes
- Erreur de configuration de l'orchestrateur

Conteneurs

- Vulnérabilités liées au temps d'exécution
- Accès illimité des conteneurs au réseau
- Configurations non sécurisées
- Vulnérabilité des applications dans les conteneurs
- Conteneurs non programmés au moment de l'exécution

Système d'exploitation hôte

- Grande surface d'attaque
- Noyau de système d'exploitation commun à tous les conteneurs
- Vulnérabilité des composants du système d'exploitation
- Mauvaise configuration des droits d'accès de l'utilisateur
- Possibilité pour les conteneurs d'accéder au système de fichiers

Exemples d'incidents liés à la sécurité des conteneurs*

	Incident	Résultat	Société
Images	Publication d'images malveillantes dans la communauté	Le temps que la vulnérabilité soit comblée (30 jours), jusqu'à 90 000 dollars ont été volés à l'aide d'images	Dépot d'image public
Registre d'images	Intégration du registre d'images dans le domaine public	Exploitation potentielle d'une vulnérabilité pour accéder à des données personnelles	Compagnie aérienne
Orchestrateur	Infiltration du système via la console d'administration non sécurisée de K8	• Minage de cryptomonnaies sur les serveurs • Fuite d'informations sensibles	Constructeur automobile
Conteneurs	Pénétration par une API non privée et exécution d'un conteneur avec des droits privilégiés	Obtention d'un accès à n'importe quel appareil du réseau	Entreprise de recherche en cybersécurité
Système d'exploitation hôte	Pénétration du système en attribuant des privilèges d'administrateur à l'hôte Docker	Obtention d'un accès à des systèmes sans serveur	Entreprise de cybersécurité

* Selon les données publiques disponibles

[2] https://www.redhat.com/en/resources/kubernetes-adoption-security-market-trends-overview

[3] https://www.redhat.com/en/blog/state-kubernetes-security-2022-1

6. Sécurité des conteneurs et des machines virtuelles

Les solutions de sécurité traditionnelles protègent les machines virtuelles, mais pas les plateformes de conteneurs.

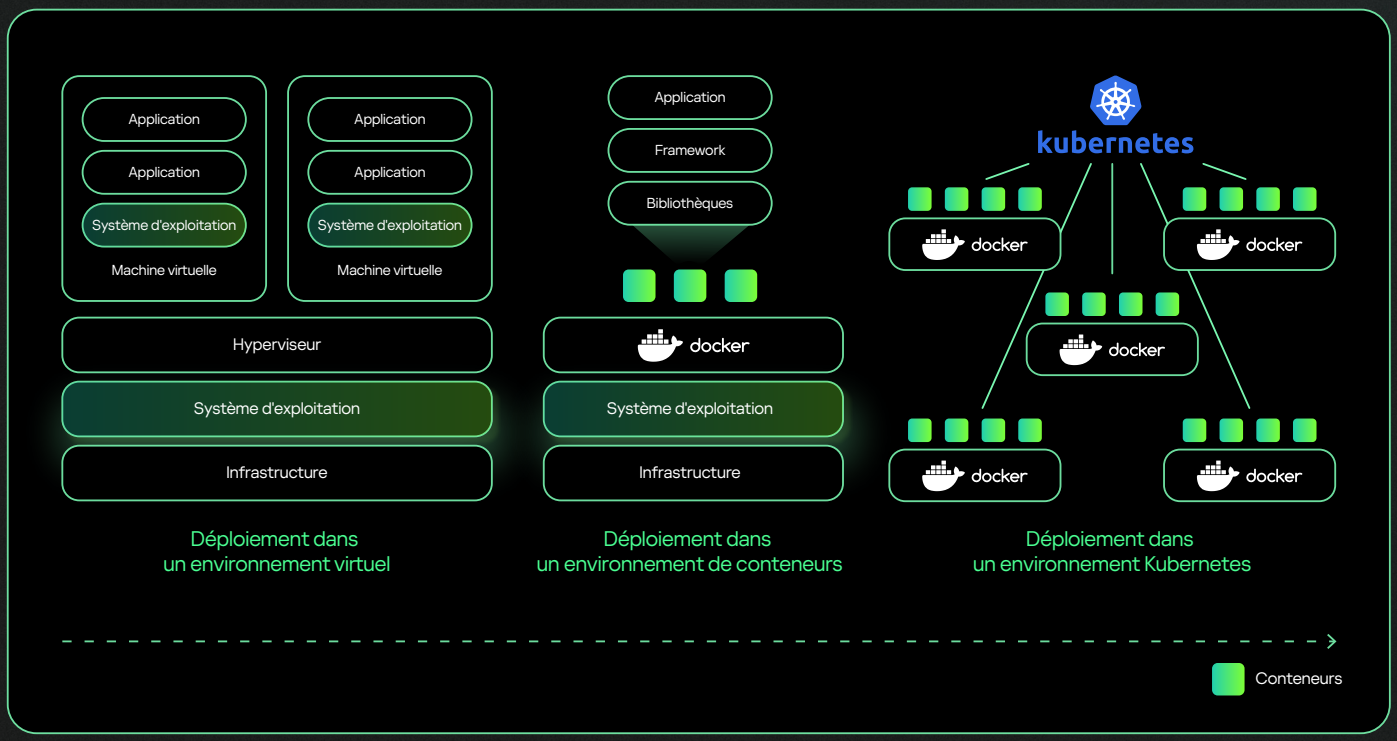
La sécurisation d'une machine virtuelle est fondamentalement la même que celle d'un ordinateur physique. Un agent de sécurité peut être facilement installé sur le système d'exploitation de la machine virtuelle.

Contrairement aux machines virtuelles, les conteneurs ne disposent pas de leur propre système d'exploitation et s'appuient sur le système d'exploitation hôte. Il est donc impossible d'installer l'agent de sécurité d'une machine virtuelle dans un conteneur. Par conséquent, les solutions de sécurité traditionnelles présentent de sérieuses limites du point de vue de la sécurité des conteneurs :

- pas d'analyse ni de contrôle des erreurs de configuration des plateformes de conteneurs.
- pas d'intégration dans les processus de développement, avec les registres d'images, les CI/CD et les plateformes d'orchestration.
- manque de visualisation des modules des environnements mis en conteneur.

L'architecture particulière des conteneurs requiert une solution de sécurité adaptée.

- Les conteneurs partagent le même noyau et les mêmes bibliothèques que le système d'exploitation hôte. Il est possible de trouver des vulnérabilités de sécurité dans les images, les registres d'images, les orchestrateurs, les conteneurs ou encore dans le système d'exploitation hôte.
- Un conteneur compromis ouvre une voie d'attaque vers tous les conteneurs partageant le noyau et le système d'exploitation hôte. Un système d'exploitation hôte compromis ouvre une voie d'attaque vers ses conteneurs.
- Les solutions de sécurité pour les machines virtuelles ne protègent pas et ne détectent pas les problèmes aux premiers stades de l'interaction avec les conteneurs : stockage, développement, déploiement.
- Les solutions de sécurité pour les machines virtuelles ne permettent pas d'analyser ni de contrôler les erreurs de configuration des plateformes de conteneurs.
- La technologie des conteneurs évolue rapidement, et les outils d'aide à la sécurité sont moins bien adoptés que pour les machines virtuelles.



7. Kaspersky Container Security

Tirez parti des possibilités et atténuez les risques courants à tous les stades (développement et mise en œuvre) grâce à Kaspersky Container Security (KCS), solution conviviale, légère et dédiée :

- Sécurise tous les composants des plateformes de conteneurs : images, registres d'images, orchestrateurs, conteneurs et système d'exploitation hôte.
- Assure une intégration transparente dans les processus de développement CI/CD sécurisés.
- Assure la protection pendant les phases de stockage et de construction, ainsi que la protection pendant l'exécution.
- Automatise l'analyse de la configuration et le contrôle de la mise en conformité.

Kaspersky Container Security est composé de trois éléments permettant de résoudre tous les problèmes de sécurité des environnements conteneurisés :

KCS Agent

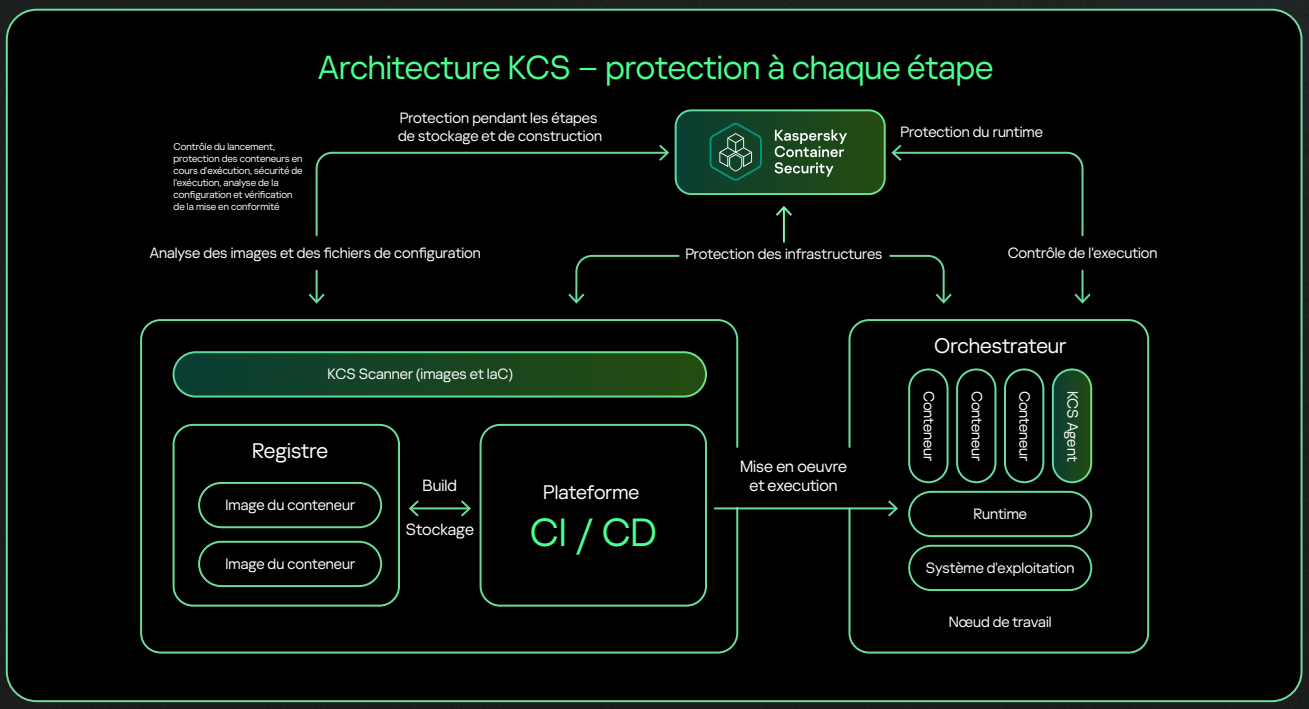
- Détecte les vulnérabilités au niveau des conteneurs, des clusters et des orchestrateurs, garantissant ainsi la sécurité de l'exécution.
- S'installe dans le cluster en tant que conteneur autonome.

KCS Scanner – analyse d'images et d'infrastructures

- Vérifie la pertinence et la sécurité du registre d'images.
- L'analyseur vérifie également les images dans le cadre du processus d'intégration continue, réduisant ainsi les risques liés à l'étape de développement.
- S'installe dans le cluster avec les composants serveur de l'orchestrateur.

KCS Control Server

- Contrôle l'état des composants de la solution et l'interaction entre eux, ainsi que l'agrégation des informations sur les événements détectés.
- S'installe dans le cluster avec les composants serveur de l'orchestrateur.
- Contrôle la transmission des images et vérifie le respect de la politique de sécurité.
- Intégration avec SIEM et d'autres solutions tierces.



Intégration dans le processus de développement

VCS et registre

L'étude

Vérifie les images à partir d'un registre
Analyse les fichiers de configuration (IaC, Dockerfile) à la recherche d'erreurs et d'informations confidentielles

Outils d'intégration continue (CI)

Code et test

Analyse les images à la recherche de vulnérabilités, de programmes malveillants et d'informations confidentielles

Outils de déploiement continu (CD)

Livraison et déploiement

Assure la diffusion d'images conformes aux politiques de sécurité

Orchestrateur

Runtime

Vérifie l'interaction avec le réseau pour s'assurer de la mise en conformité avec les stratégies de sécurité
Analyse comportementale des conteneurs

Avantages de Kaspersky Container Security

- Optimise les coûts de la protection de l'environnement des conteneurs.
- Facile à utiliser, ce qui réduit la charge de travail de l'équipe de sécurité.
- Comprend toutes les fonctionnalités et possibilités correspondant aux bonnes pratiques mondiales en matière de sécurité des conteneurs.
- Respecte les exigences réglementaires et les normes de sécurité de l'industrie.
- Intégration avec SIEM et d'autres solutions tierces.

Atténue les risques liés aux composants clés dans les environnements conteneurisés

Images

Vérification des vulnérabilités

Vérifie les erreurs dans les configurations d'images

Détection de programmes malveillants

Vérifie la présence d'informations confidentielles

Évaluation des risques et identification des images potentiellement dangereuses

Registre d'images

Intégration aux registres et validation des images conformément aux stratégies d'analyse

Utilise des images sécurisées à jour

Orchestrateur

Détecte les erreurs de configuration et fournit des recommandations pour les corriger

Visualise les ressources d'un cluster

Surveillance du trafic

Découvre et analyse les images d'un cluster

Conteneurs

Contrôle le lancement et le fonctionnement des conteneurs de confiance uniquement

Surveillance du trafic

Contrôle de l'intégrité des conteneurs

Protection contre les menaces ciblant les fichiers

Analyse comportementale

Contrôle le fonctionnement des applications et des services à l'intérieur des conteneurs

Système d'exploitation hôte

Vérifie les versions des modules du système d'exploitation de base

Détecte les erreurs de configuration et fournit des recommandations pour les corriger

Réduit le risque en surveillant les conteneurs

Si vous souhaitez en savoir plus, visitez notre site Web.

Découvrez le fonctionnement

www.kaspersky.fr