12

règles immuables de

l'observabilité

splunk>

turn data into doing™

Introduction

La vitesse est synonyme de succès dans l'économie numérique d'aujourd'hui. Les clients attendent des expériences numériques irréprochables et la concurrence n'est toujours qu'à un clic de souris : les entreprises se tournent donc vers des technologies natives du cloud comme les microservices, les conteneurs et Kubernetes pour accélérer l'innovation, créer des applications plus rapidement et améliorer leurs performances. Toutefois, le passage à des technologies natives du cloud et à des architectures distribuées pose de nouveaux défis en matière de vitesse, d'échelle et de complexité des données, défis que les solutions de supervision traditionnelles ne sont tout simplement pas conçues pour gérer.

C'est là que l'observabilité entre en jeu.

Les entreprises doivent livrer un code de grande qualité et des expériences utilisateur différenciées, sans perdre une seconde. L'observabilité permet aux équipes DevOps et SRE de comprendre et d'expliquer les comportements inattendus, afin de gérer efficacement et proactivement les performances des microservices distribués exécutés sur une infrastructure éphémère. Une stratégie et une solution d'observabilité appropriées se traduisent par une plus grande fiabilité, une meilleure expérience client et une productivité accrue.

Nous proposons ici 12 règles immuables de l'observabilité visant à pérenniser votre réussite, quelle que soit la complexité de votre environnement. Plus un système est observable, plus vite nous pouvons comprendre la cause des problèmes et les corriger : un facteur essentiel pour atteindre les indicateurs et les objectifs de niveau de service (SLI et SLO) et, en fin de compte, pour accélérer les résultats commerciaux.

Mais tout d'abord, une définition rapide de l'observabilité :

L'observabilité indique dans quelle mesure nous pouvons déduire l'état de nos systèmes (infrastructure, services, etc.), ou répondre à toute question les concernant, à l'aide de leurs données de télémétrie (métriques, traces et logs).

Les systèmes observables permettent aux équipes DevOps de résoudre tout problème pouvant survenir dans leurs systèmes, y compris les défaillances inconnues et celles dont les causes profondes sont enfouies dans un labyrinthe de microservices.

Au-delà de la résolution des problèmes, l'observabilité permet aux équipes d'accéder à leurs systèmes et d'améliorer de manière proactive les publications de code et l'architecture du système, et de s'adapter au changement plus rapidement.

Cependant, la disponibilité des données ne constitue pas à elle seule une solution d'observabilité. À mesure que l'observabilité devient une partie intégrante de la chaîne d'outils DevOps, il est important de garder à l'esprit certaines règles immuables lors du choix, de l'adoption et de l'amélioration de votre solution d'observabilité.

L’observabilité commence par les données

Comment savoir *vraiment* ce que font vos services, même au cours du développement ? Tout commence par les données.

01 Une solution d'observabilité utilise toutes vos données pour éviter les angles morts

Le seul moyen de résoudre une défaillance inconnue profondément enfouie et d'optimiser le comportement d'une application consiste à mesurer et à recueillir toutes les données relatives à votre environnement, en haute-fidélité et sans échantillonnage. C'est le seul moyen de garantir l'absence de lacunes dans la visibilité. Vous disposez des données dont vous avez besoin, quand vous en avez besoin.

Les architectures distribuées orientées services créent des interactions, des dépendances et des propagations d'erreurs plus complexes d'un service à l'autre, produisant des systèmes très imprévisibles caractérisés par une longue traîne de problèmes peu fréquents mais graves. Les solutions d'observabilité traditionnelles sont rarement adaptées à la supervision des applications basées sur les microservices, car elles appliquent un échantillonnage probabiliste qui prélève les traces de manière aléatoire et néglige souvent celles qui vous intéressent (transactions uniques, anomalies, valeurs aberrantes, etc.).

Lorsque vous évaluez des solutions d'observabilité, recherchez celles qui ne font pas d'échantillonnage et qui conservent toutes vos traces (vous pouvez choisir celles que vous souhaitez conserver), et qui remplissent les tableaux de bord, les cartes de service et les navigateurs de trace avec des informations utiles qui vous aideront réellement à superviser et à dépanner votre application.

02 Fonctionne à la vitesse et à la résolution de votre nouvelle infrastructure définie par logiciel (ou cloud)

Des cas d'utilisation différents présentant des degrés de priorité variables nécessitent des résolutions différentes. La résolution à laquelle vous collectez des données de votre application monolithique sera certainement insuffisante lorsque vous commencerez à collecter les données de microservices plus dynamiques s'exécutant sur des conteneurs éphémères et des fonctions serverless. Par exemple, si vous mesurez les performances d'une application monolithique mature exécutée sur des machines virtuelles (VM) surprovisionnées avec un nombre relativement constant d'utilisateurs, une visibilité relativement grossière (résolution à l'échelle de la minute) sur votre infrastructure peut suffire. En revanche, si vous disposez de microservices qui s'exécutent sur des conteneurs éphémères, orchestrés par Kubernetes et mis en service ou hors service automatiquement en quelques minutes, ou de fonctions serverless qui s'instancient pendant quelques secondes seulement, vous aurez besoin d'une granularité beaucoup plus fine (résolution d'une seconde) pour superviser efficacement les performances de votre application et de votre infrastructure.

Lorsque vous commencerez à adopter des microservices, il vaudra mieux s'appuyer sur une résolution trop élevée que trop faible, car le processus de réarchitecture d'une application ou de création d'une nouvelle application réseau « cloud-native » implique souvent un processus par tâtonnements

En d'autres termes, vous avez besoin d'une observabilité qui fonctionne à la même échelle de temps que votre infrastructure éphémère définie par logiciel.

03 Exploite des instruments ouverts et flexibles et facilite l'utilisation par les développeurs

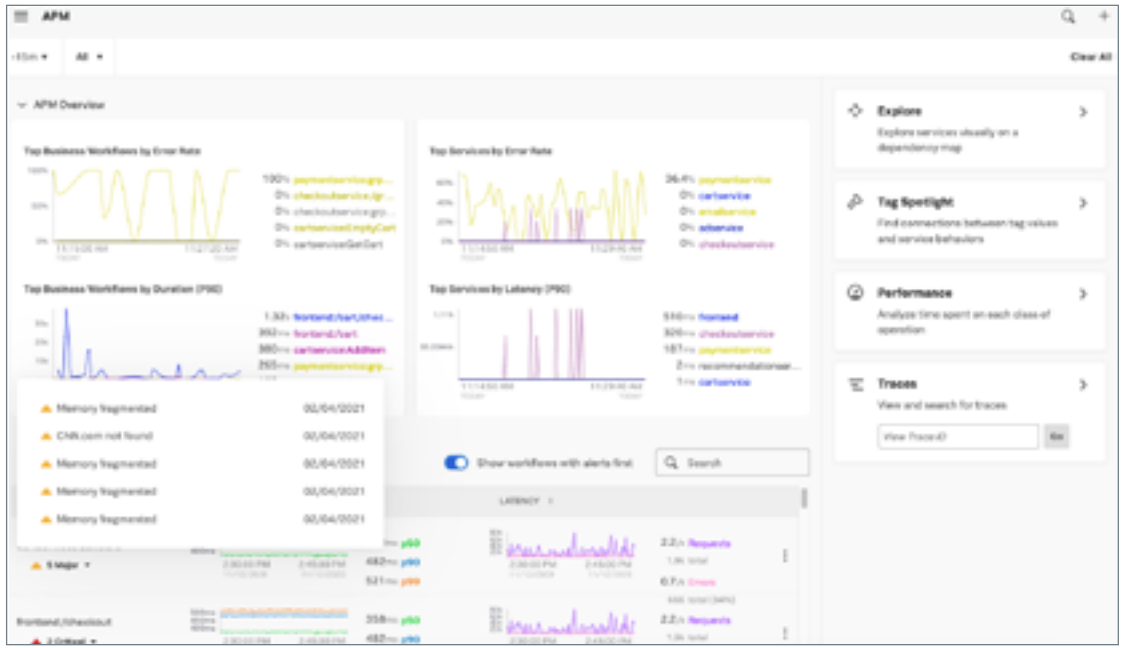
Prévoyez d'utiliser une méthode de collecte de données ouverte standard dès le premier jour. En sélectionnant un format de données normalisé pour les données de trace, de métriques et de log, et en optant pour des méthodes d'acquisition de données ouvertes, vous pourrez plus facilement instrumenter votre code et commencer à capturer des données d'observabilité, ce qui fera gagner du temps aux développeurs.

Les agents lourds et propriétaires sont difficiles à entretenir, dégradent les performances des services et sont difficiles à remplacer : ils produisent une solution d'observabilité spécifique incapable de répondre à l'évolution de vos besoins et qui risque de devenir de plus en plus coûteuse au fil du temps. En choisissant de s'appuyer sur des langages et des frameworks communs et en tirant parti d'OpenTelemetry, le deuxième projet le plus actif de la Cloud Native Computing Foundation, vous bénéficiez d'une flexibilité optimale, non seulement pour la collecte des données, mais également dans le choix des solutions cloud que vous allez utiliser. Plus important encore, l'utilisation d'OpenTelemetry met toutes les chances de votre côté lorsque vous devrez évoluer et étendre la supervision et le dépannage à un nombre toujours croissant de microservices distribués.

Une instrumentation ouverte facilite également l'intégration avec l'ensemble de votre chaîne d'outils existante pour une visibilité totale

du code au cloud. L'observabilité offre une visibilité sur vos pratiques DevOps et votre chaîne d'outils. Cette visibilité est essentielle au maintien de la vélocité des applications et à la prise en charge de nouveaux outils et processus. Bien qu'il n'existe pas de « taille unique » en matière de solution, de langage, de produit de réponse aux incidents ou d'arsenal DevOps pour le cloud, votre solution d'observabilité doit être en mesure de s'intégrer facilement à tous les outils que vous utilisez ou que vous pourriez utiliser à l'avenir, et de fournir des informations sur ceux-ci.

04 Crée un workflow transparent pour la supervision, le dépannage et la résolution en corrélant les données et en tissant des liens entre les métriques, les traces et les logs



Les organisations gèrent de nombreux outils spécialisés. Il n'est pas rare de voir les responsables d'applications signaler une dégradation des performances à l'aide d'un outil APM ponctuel, puis contacter une autre équipe des opérations IT bénéficiant d'une vue distincte sur les serveurs et les hôtes grâce à un outil ponctuel de supervision de l'infrastructure, pour tenter de déterminer si des problèmes d'infrastructure affectent les applicatifs critiques et les performances de l'entreprise.

Cette approche n'est pas viable lorsque l'on tente de résoudre les problèmes rapidement. Vous devez être en mesure de comprendre facilement les interdépendances et tout effet en amont ou en aval d'un problème particulier dans un système, et disposer d'un workflow cohérent quel que soit l'endroit où commence l'investigation.

Votre solution d'observabilité doit être entièrement intégrée pour offrir des informations contextuelles pertinentes tout au long du processus de résolution des problèmes, quel que soit votre poste (ingénieur front-end, SRE, ingénieur DevOps). Suivez la trace des données sans tomber dans

une impasse. Par exemple, il devrait être facile de passer d'une alerte aux détails de trace correspondants et de transmettre ces informations pour mettre en corrélation la dégradation des applications avec des problèmes de l'infrastructure sous-jacente ou du front-end.

De nombreuses équipes créent et proposent des applications modernes, ce qui rend la résolution des problèmes difficile et cloisonnée. L'observabilité permet de réduire les étapes de découverte inutiles et les fausses pistes de dépannage en assurant une circulation transparente des informations pour un flux d'investigation continu.

Fig. : déclenchement de métriques RED

Ne contraignez pas vos équipes à répéter les étapes d'une investigation. Pourquoi les métriques, les traces et les logs contextuels et unifiés jouent-ils un rôle clé dans un workflow de dépannage :

Une alerte sur un service (métrique)	SRE, opérations
Entraîne une erreur de temporisation (trace)	ingénieur DevOps, ingénieur logiciel
Entraîne un problème d'infrastructure (métrique)	ingénieur DevOps, SRE
Entraîne un problème de configuration (métrique)	ingénieur DevOps, opérations
Entraîne une fuite de mémoire dans une application (logs)	développeur, ingénieur logiciel

L’observabilité permet d’obtenir des réponses rapidement

Les architectures modernes s’accompagnent d’une vague de données qui a un impact sur la compréhension de vos systèmes. Mais les données à elles seules n’ont pas de sens : vous devez les agréger, les analyser et y répondre au besoin.

05 Facilite l'utilisation, la visualisation et l'investigation des données dès le départ

Des visualisations intuitives ne nécessitant aucune configuration, comme les tableaux de bord, les graphiques et les cartes thermiques, permettent de comprendre en un regard les quantités considérables de données que vos systèmes produisent et vous permettent d'interagir avec les mesures clés en temps réel. Assurez-vous que votre solution d'observabilité agrège toutes les données, affiche automatiquement des tableaux de bord des métriques, des cartes des services et les architectures de conteneurs, et permet d'effectuer un filtrage, un regroupement et une agrégation dynamiques selon différentes dimensions. Votre solution doit également permettre de créer des tableaux de bord personnalisés pour garder un œil sur certains services d'intérêt.

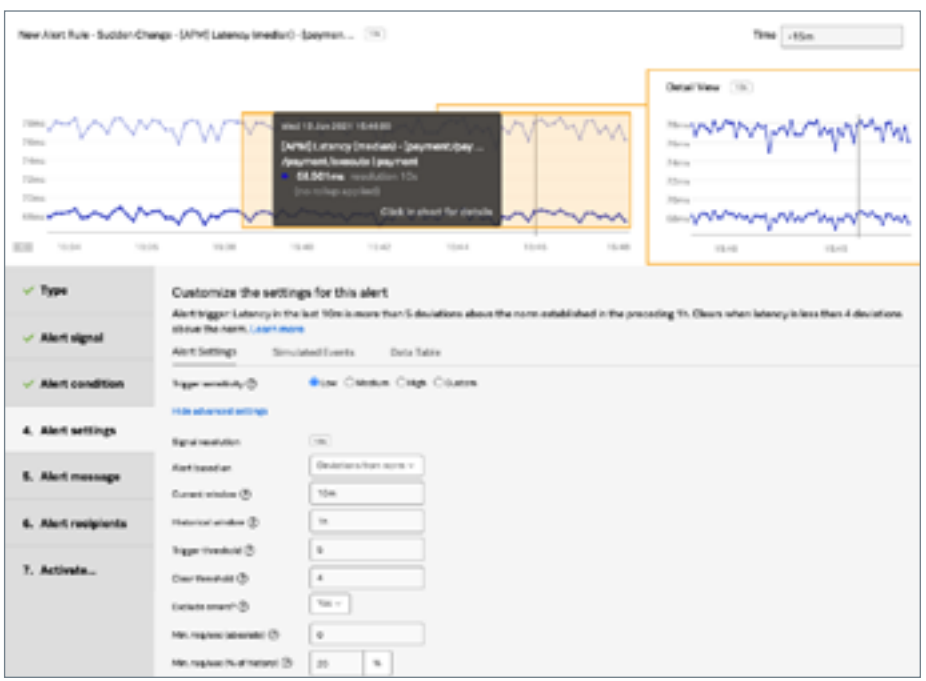
Comme nous l'avons dit, le contexte est essentiel. Imaginez que votre solution d'observabilité vous informe que la latence du 99e centile de votre service a augmenté. Vous suivez un lien vers le tableau de bord du service, directement à partir de la fenêtre d'alerte. Le tableau de bord du service affiche tous ses composants et les graphiques indiquent un problème au niveau du magasin de données. Vous suivez le lien vers le tableau de bord du magasin de données, et en effet, il y a environ 15 minutes, l'une des instances a commencé à afficher un pic de latence. Vous savez maintenant exactement où et quand le problème a

commencé. Armé du quoi, du où et du quand, vous pouvez maintenant suivre un lien vers les logs pour découvrir pourquoi, par exemple en examinant les traces de la pile complète inscrites dans les logs.

Votre outil d'observabilité doit vous faciliter la tâche : une alerte, deux tableaux de bord et trois clics pour atteindre la source du problème.

06 Tire parti de l'intelligence artificielle sur le flux pour produire des alertes plus rapides et plus précises, proposer un dépannage guidé et offrir des informations rapidement

La quantité de données produites par les environnements natifs du cloud est trop importante pour être interprétée manuellement. Pour traiter rapidement toutes ces données, il vous faut des analyses en temps réel qui vont mettre en lumière les motifs pertinents et fournir de manière proactive des informations exploitables. Les alertes de base, déclenchées par des seuils statiques et des prises de pouls, sont souvent imprécises et génèrent beaucoup de bruit. Elles provoquent souvent des déluges d'alertes qui frustrent les ingénieurs d'astreinte et ajoutent au problème au lieu de contribuer à le résoudre. Au lieu de continuer à dépendre de ces conditions d'alerte inefficaces, envisagez des seuils plus dynamiques basés sur des modèles statistiques avancés et l'IA, ainsi que des règles plus complexes et multi-conditions. Recherchez une solution capable de mesurer efficacement les performances historiques, d'effectuer des comparaisons sophistiquées et de détecter les valeurs aberrantes et les anomalies en temps réel. Elle doit également vous permettre d'ajuster et de personnaliser vos règles d'alerte en fonction de votre environnement d'application spécifique.



07 Fournit un feedback rapide sur les modifications (de code), même en production

L'observabilité ne se limite pas aux opérations et doit être intégrée dès le développement.

Le décalage vers la gauche, qui consiste à intégrer des processus DevOps comme les tests plus tôt dans le pipeline, devient une stratégie populaire pour les équipes qui cherchent à localiser et à corriger les problèmes plus rapidement.

Le décalage vers la droite, qui implique l'extension des processus pré-déploiement à la phase de production du pipeline, permet d'élargir la couverture des tests et de la supervision.

Une fois le code déployé, les équipes doivent comprendre ce qui se passe au sein de leurs applications, au fil de chaque nouvelle publication dans le pipeline de distribution. Si vous ne comprenez pas ce qui se passe à l'intérieur de votre application, vous ne pouvez pas comprendre votre pipeline ni corréler les événements du pipeline avec les performances de l'application et l'expérience de l'utilisateur final.

C'est là que les tests d'applications et la gestion des performances entrent en jeu pour offrir une visibilité totale du code au cloud.

L'observabilité propose une supervision synthétique, l'analyse des transactions des utilisateurs réels, l'analyse des logs et le suivi des métriques, permettant aux équipes de comprendre l'état de leur code, du développement au déploiement. Cette compréhension offre la profondeur dont les équipes ont besoin pour obtenir une visibilité sur l'état de chaque version, tout au long du cycle de vie du développement.

Splunk pour les DevOps

08 Automatise et vous permet d'en faire un maximum « en tant que code »

Améliorez considérablement la productivité, l'efficacité et la prévisibilité de votre organisation en exploitant la programmabilité partout où c'est possible. Exploitez les API pour gérer automatiquement les ressources de l'infrastructure (par exemple, via un orchestrateur Kubernetes), le contrôle des modifications et les déploiements de code (par exemple, via l'intégration Jenkins). Créez une automatisation en boucle fermée dans l'ensemble de votre environnement de production pour déclencher, à partir d'alertes en temps réel, des opérations sophistiquées telles que des rétrogradations et des mesures correctives automatiques, pour réduire le temps moyen de résolution (MTTR) à la vitesse de la machine.

```
provider "signalfx" {
   # It is strongly recommended to use secret management Terraform Provider such as Vault
   auth_token="<<your access token>>"
   api_url = "https://api.us1.signalfx.com" #use your custom SignalFx URL
}
resource "signalfx_detector" "application_latency" {
 name        = "application latency is high"
 description = "SLI metric for application latency is higher than expected."
 program_text = <<-EOF
       signal = data('demo.trans.latency').max()
       detect(when(signal > 250, '1m')).publish('application latency is greater than 250 ms')
   EOF
 rule {
   description   = "Application latency was high for last one minute"
   severity      = "Warning"
   detect_label  = "application latency is greater than 250 ms"
   notifications = ["Email,amitsharma@splunk.com"] #you can also configure slack, VictorOps and others
 }
}
```

Fig. : exemple de création d'un détecteur d'alerte dans Splunk à l'aide du fournisseur Terraform

Et dans le mouvement « tout en tant que code », l'observabilité ne fait pas exception. Selon le principe de « l'observabilité en tant que code », vous développez, déployez, testez et partagez des actifs d'observabilité tels que des détecteurs, des alertes, des tableaux de bord, etc. sous la forme de code.

La supervision et les alertes en tant que code impliquent d'intégrer la création et la maintenance automatisées de graphiques, de tableaux de bord et d'alertes dans le cycle de vie des services. En procédant de cette façon, les visualisations et les alertes sont toujours à jour, sans prolifération, et vous pouvez maintenir le contrôle des versions via un dépôt centralisé, sans avoir à gérer chaque composant manuellement en permanence. L'utilisation des API disponibles et de la programmabilité permet également de garantir la conformité générale des visualisations et des alertes aux bonnes pratiques et aux politiques de l'entreprise.

L’observabilité est essentielle à votre culture et à votre stratégie commerciale

L’observabilité est un investissement vital pour l’entreprise, surtout lorsque quelques secondes d’arrêt peuvent coûter des millions de dollars. Elle dépasse le champ des équipes DevOps pour appuyer la résilience et une expérience client irréprochable.

09 Est un élément essentiel de la mesure des performances de l'entreprise

La fiabilité est essentielle lorsqu'il s'agit de fournir des applications hautes performances et des expériences client irréprochables, mais il n'y a pas de fiabilité sans observabilité. Sans observabilité, comment savoir où investir du temps et des ressources ? Si vous ne mesurez pas la disponibilité, comment connaître votre fiabilité ? Si vous ne mesurez pas les performances, comment évaluer la qualité de ce que vous proposez ? Ces mesures doivent s'appliquer à tout le cycle, du développement à la production. À l'ère des données, vous devez avoir de la visibilité sur chaque étape de la livraison.

L'observabilité ouvre une fenêtre qui permet de voir plus loin que la consommation de CPU et les mesures de base dans chaque couche de la pile, et produit des informations sur l'expérience utilisateur, les performances SLX et d'autres indicateurs clés alignés sur vos besoins commerciaux. Dans les environnements natifs du cloud, des pics mineurs dans un service peuvent se traduire par une augmentation de la latence, parfois même pour un client spécifique.

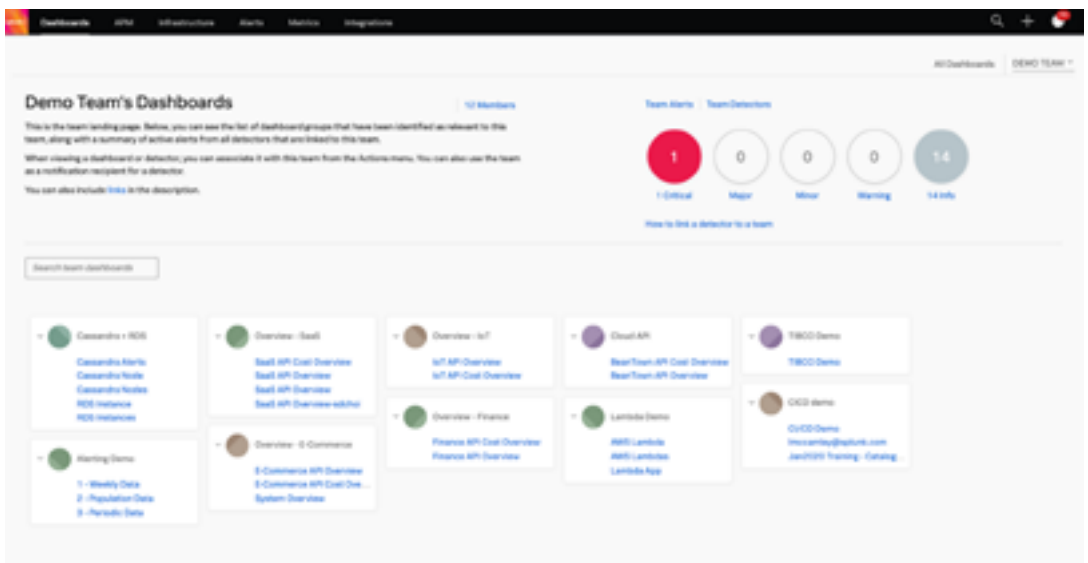
Il est important de comprendre les KPI qui servent à mesurer votre activité et de quelle façon les équipes de votre entreprise vont consommer les données. Vous allez pouvoir :

- anticiper les dimensions nécessaires à vos données de supervision ;
- corréler les données de toute votre pile, de l'infrastructure sous-jacente à vos applications et vos microservices ;
- corréler les données sur l'ensemble de votre activité numérique.

Pour prendre un exemple simple, imaginons une application qui s'exécute sur AWS et qui a des utilisateurs dans le monde entier. Pour obtenir une vue d'ensemble de l'expérience de l'utilisateur final dans chaque région du monde, il faudra pouvoir répartir les utilisateurs, les microservices et l'infrastructure en fonction de la région ou de la zone de disponibilité AWS. Savoir à l'avance de quel type de métadonnées vous avez besoin vous aidera à configurer vos visualisations correctement dès la première fois.

10

Fournit l’observabilité en tant que service



Le paradigme DevOps selon lequel « si vous le développez, vous le gérez », encourage l’agilité notamment en décentralisant la responsabilité opérationnelle vers des équipes individuelles. De plus en plus de personnes dans l’entreprise ont désormais besoin d’un accès à l’observabilité, et cette décentralisation peut facilement conduire à la fragmentation des outils et des données. La fragmentation peut entraîner des coûts plus élevés et, pire encore, de grandes pertes d’efficacité dans les opérations. Et comme les ressources cloud sont illimitées, les coûts peuvent être encore plus difficiles à gérer.

Les plateformes modernes d’observabilité offrent une gestion centralisée, ce qui permet aux équipes et aux utilisateurs de disposer de contrôles d’accès et de gagner en transparence et en contrôle sur la consommation. La mise en œuvre de bonne pratiques claires d’observabilité dans l’ensemble de votre entreprise peut non seulement offrir une meilleure expérience aux développeurs, leur permettre de travailler plus efficacement et de se concentrer sur la création de nouvelles fonctionnalités, mais également améliorer la collaboration inter-équipes, l’évaluation des coûts et les performances globales de l’entreprise.

11

Intègre en toute fluidité la collaboration, la gestion des connaissances et la réponse aux incidents

Les incidents sont inévitables, mais une solution d'observabilité solide peut limiter les temps d'arrêt, voire les empêcher entièrement, pour réduire les coûts de l'entreprise et améliorer la qualité de vie des ingénieurs d'astreinte. Mais la plupart des entreprises ne savent pas qu'elles ont du pouvoir sur la préparation du rétablissement. Pour répondre aux problèmes et les résoudre rapidement (en particulier dans un environnement de déploiement à grande vitesse), vous aurez besoin d'outils qui facilitent une collaboration efficace et des signalements rapides. Les solutions d'observabilité doivent inclure des capacités de réponse automatisée aux incidents afin d'impliquer le bon expert au bon moment, et ainsi réduire considérablement les temps d'arrêt.

D'autres bonnes pratiques à prendre en compte dans la poursuite de votre objectif d'observabilité :

- abandonnez la division tribale des connaissances et l'héroïsme individuel dans la résolution des problèmes au profit d'une approche normalisée reposant sur des procédures et des bases de connaissance. Fournir à tous les ingénieurs d'astreinte un accès facile à du contexte détaillé et des suggestions de résolution de problèmes similaires antérieurs représente un aspect fondamental du partage d'informations, de la collaboration et de la réduction du MTTR ;
- permet un accès transparent aux outils tiers (pour le suivi des erreurs, notamment) via des liens web faciles à utiliser pour vos ingénieurs d'astreinte. Ils peuvent ainsi facilement importer le contexte de l'incident dans d'autres systèmes et poursuivre leur procédure de dépannage sans perdre une seconde.

12 Peut évoluer pour soutenir la croissance future avec élasticité

Investissez en fonction de vos besoins futurs en observabilité plutôt que ceux d'aujourd'hui. Combien de conteneurs avez-vous ? Combien avez-vous d'hôtes dans votre environnement, d'applications en production et de publications de code chaque jour ? Et chaque année ? Répondez à ces questions et vous comprendrez pourquoi vous avez besoin d'un système de supervision évolutif (ou en aurez bientôt besoin). Pour répondre aux besoins de n'importe quel environnement, quelle que soit sa taille ou sa complexité, les solutions d'observabilité doivent être capables d'importer des pétaoctets de données de log et des millions de métriques et de traces, tout en maintenant des performances élevées. Vous garantirez ainsi la pérennité de votre investissement.

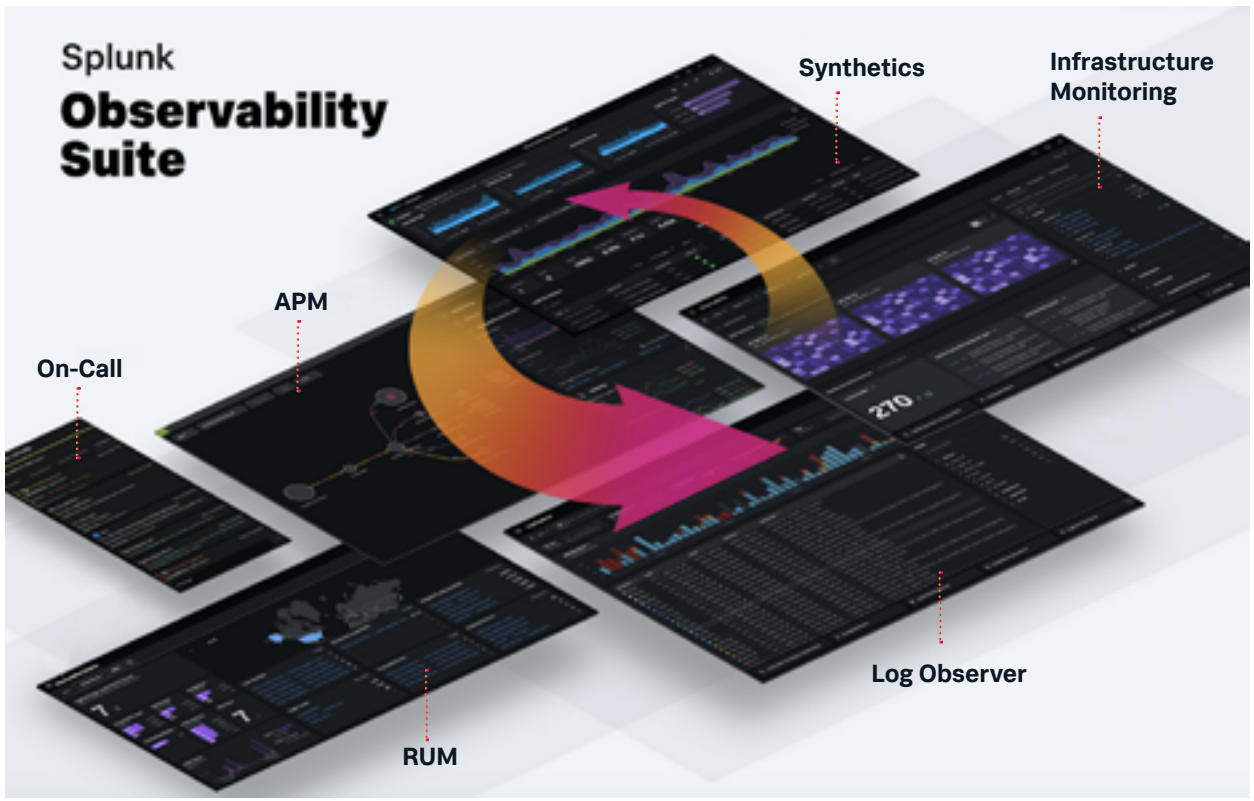
Développer votre stratégie d'observabilité avec Splunk

L'observabilité est essentielle à la réussite, mais elle fait rarement partie des compétences de base d'une entreprise. C'est pourquoi il est important de collaborer avec un fournisseur de solutions d'observabilité comme Splunk qui peut vous soutenir dans votre parcours vers le cloud.

La Splunk Observability Suite offre aux utilisateurs un workflow fluide et harmonisé pour la supervision, le dépannage et l'investigation, qui permet de passer facilement de la détection des problèmes à leur résolution en quelques minutes. Que vous soyez un développeur front-end qui a besoin de comprendre l'expérience des clients, un développeur back-end qui élabore des API et des services ou un SRE fréquemment d'astreinte, la Splunk Observability Suite vous aide à obtenir les informations dont vous avez besoin et à collaborer avec les personnes qui peuvent résoudre rapidement les interruptions de service.

La Splunk Observability Suite offre :

- une expérience utilisateur étroitement intégrée, basée sur des workflows transparents et riches en contexte pour la supervision, le dépannage et l'investigation ;
- une visibilité de bout en bout basée sur l'acquisition ouverte et la corrélation de TOUTES les données, métriques, traces et logs ;
- une vitesse et une évolutivité inégalées grâce à un moteur d'analyse de flux pour détecter les problèmes et obtenir des retours en temps réel (en quelques secondes plutôt qu'en plusieurs minutes) ;
- des analyses basées sur IA qui tirent parti de l'ensemble des données et fournissent des informations exploitables.



En savoir plus

Pour en savoir plus sur la Splunk Observability Suite, visitez notre site web et regardez notre démonstration.

Regarder la démonstration

21-17608-SPLK-12-Immutable-Rules-for-O11y-107

splunk> turn data into doing™