Engineer

Indisponibilité

🛟 période où tout ou partie d'un service est inutilisable

Ce que c’est

L’indisponibilité est le temps pendant lequel un service ne fonctionne pas pour les personnes qui en dépendent. On parle aussi d’interruption ou de panne. Son contraire est le temps de fonctionnement ; la part du temps pendant laquelle un service fonctionne est sa disponibilité, généralement exprimée en pourcentage sur un mois ou une année.

Le cas simple est une interruption totale : le site ne se charge pas, l’API refuse les connexions, la base de données est hors service. Le cas courant est moins net. Un passage en caisse qui échoue faute de réponse dans le délai imparti alors que le catalogue fonctionne, une page de connexion qui se charge mais rejette tous les mots de passe, une API qui ne répond qu’à une requête sur trois : chacun constitue une indisponibilité pour la personne qui a besoin de ce parcours, même si un contrôle de bon fonctionnement visant la page d’accueil reste au vert. Une définition utile compte ce que les utilisateurs peuvent faire, plutôt que de vérifier seulement si un processus tourne.

Il existe deux types d’indisponibilité. L’indisponibilité non planifiée est l’interruption que personne n’a choisie : une panne de disque, un mauvais déploiement, un certificat expiré, une dépendance tombée en panne en premier. L’indisponibilité planifiée est décidée à l’avance, généralement dans une fenêtre de maintenance, pour mettre à niveau, migrer ou déplacer un élément qui ne peut pas changer tout en servant du trafic. Elle est plus facile à supporter, puisque les utilisateurs peuvent être avertis, mais elle réduit tout de même la disponibilité, sauf exclusion explicite dans l’accord.

HTTP possède un vocabulaire pour l’interruption elle-même. La RFC 9110 définit la classe de codes d’état 5xx pour un serveur qui sait avoir commis une erreur ou ne pas pouvoir exécuter la demande. Trois de ces codes ont des sens distincts

pendant une interruption. 502 Bad Gateway signifie qu'un proxy a reçu une réponse invalide du serveur situé derrière lui. 504 Gateway Timeout signifie que le proxy a attendu cette réponse trop longtemps. Et 503 Service Unavailable est celui à envoyer délibérément : la spécification l'associe à une surcharge temporaire ou à une maintenance planifiée et permet au serveur d'indiquer quand revenir :

```
HTTP/1.1 503 Service Unavailable
Retry-After: 120
```

Cela indique aux navigateurs, aux clients d'API et aux robots des moteurs de recherche que la situation est temporaire et suggère d'attendre deux minutes. Le message est très différent de celui d'une page défectueuse servie avec un code de succès.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que l'indisponibilité ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Une supervision et des alertes qui signalent les incidents tant qu'ils restent mineurs, avec les chiffres qui montrent à quelle fréquence ils le restent.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. Premier tableau de bord d'observabilité — première vue de l'entreprise sur ses propres systèmes

Premier tableau de bord d'observabilité Contactez-nous →

Pourquoi c'est important

On décrit les objectifs de disponibilité par leur nombre de « neuf », et ces « neuf » se traduisent en temps. Sur une année de 365 jours, soit 8 760 heures, chaque objectif autorise cette durée d'indisponibilité :

- 99 % : 87,6 heures par an, soit environ trois jours et demi ;
- 99,9 % : 8,76 heures par an, ou 43,2 minutes sur un mois de 30 jours ;
- 99,95 % : 4,38 heures par an, ou 21,6 minutes sur un mois de 30 jours ;
- 99,99 % : 52,56 minutes par an ;
- 99,999 % : environ 5,26 minutes par an.

Chaque neuf supplémentaire divise la marge par dix, et le coût nécessaire pour l'obtenir augmente fortement. À 99,9 %, on peut réveiller quelqu'un, lui laisser le temps de se connecter, de trouver le problème et de le corriger dans le budget mensuel. À 99,99 %, l'année entière autorise moins d'une heure, ce qui laisse peut-être la place à un seul incident ; la reprise doit donc être automatique. À 99,999 %, même un redémarrage courant doit se dérouler sans que les utilisateurs s'en aperçoivent.

Les disponibilités des dépendances se multiplient. Une requête ayant besoin d'un répartiteur de charge, d'une application et d'une base de données, disponibles chacun 99,9 % du temps et tombant en panne indépendamment, n'est servie que lorsque les trois fonctionnent : $0{,}999^3$, soit environ 99,7 %, ce qui représente à peu près 26 heures d'indisponibilité par an au lieu de 8,76. Un service ne peut pas être plus disponible que la chaîne de ses dépendances, à moins d'en supprimer ou d'en doubler des maillons.

La planification commence par les besoins de l'activité. Le guide du NIST sur la planification de secours des systèmes fédéraux, SP 800-34, nomme trois chiffres à fixer avant la première panne. La durée maximale d'interruption tolérable (MTD) est la durée totale que le responsable du système accepte. L'objectif de délai de reprise (RTO) est le temps pendant lequel un système peut rester indisponible avant que les conséquences deviennent inacceptables ; il doit être inférieur à la MTD. L'objectif de point de reprise (RPO) indique jusqu'où les données peuvent être ramenées dans le passé à partir de la dernière sauvegarde. Il mesure donc une perte de données tolérable, plutôt qu'une durée d'interruption. Une fois ces trois valeurs écrites, le calendrier des sauvegardes et la conception de la bascule répondent à une question au lieu de reposer sur des suppositions.

À quoi cela ressemble en pratique

Sur une plateforme de streaming IPTV et OTT ayant des clients en Chine, aux États-Unis et en Allemagne, chaque heure était l'heure de grande écoute de quelqu'un ; il n'existait donc pas d'heure creuse pour la maintenance. L'assistance fonctionnait jour et nuit pour environ un millier de serveurs, auxquels s'ajoutait un nombre appréciable de systèmes appartenant aux clients. Une grande partie du travail consistait à repérer une dérive avant qu'elle ne devienne une interruption : sur un flux en direct, une interruption, c'est la télévision de quelqu'un qui s'arrête. La plateforme est restée disponible en continu pour son public mondial.

Dans une entreprise de taille moyenne, dix serveurs de bases de données PostgreSQL et SQL Server ont fonctionné avec une disponibilité de 99,9 %. Selon le calcul ci-dessus, cela représente une marge de 8,76 heures par an pour chacun. Derrière ce chiffre se trouvaient des sauvegardes automatisées par `pg_dump` et des tâches SQL Server Agent, avec des politiques de rétention et un stockage hors site, une supervision dans Prometheus et Grafana, ainsi que des procédures de reprise après sinistre documentées pour les restaurations à un instant donné et les bascules.

Et sur un parc de trente serveurs Ubuntu dont la configuration s'était développée au gré des circonstances, le volet reprise après sinistre a été reconstruit autour de sauvegardes et de procédures de restauration effectivement testées. La configuration réseau est devenue un socle durci commun à toutes les instances. Le risque d'indisponibilité a diminué grâce à des parcours de reprise testés avant que quiconque en ait besoin.

Où les problèmes surviennent

La supervision se trouve à l'intérieur de la panne. Un contrôle de bon fonctionnement exécuté sur le même serveur ou dans le même centre de données continue d'annoncer du vert alors que les utilisateurs rencontrent un certificat TLS expiré, un enregistrement DNS cassé ou un répartiteur de charge qui ne pointe vers rien. Sondez depuis l'extérieur, depuis plusieurs endroits et en suivant le parcours d'un utilisateur réel, puis déclenchez les alertes d'après ce que cette sonde constate.

La page d'erreur répond 200. Un framework qui affiche une page d'erreur accueillante avec un code d'état de succès dissimule l'interruption à la supervision, aux clients qui auraient réessayé et aux robots, qui risquent d'indexer la page d'erreur à la place de la vraie. Les erreurs devraient porter un code d'erreur ; une maintenance planifiée devrait répondre 503 avec un en-tête Retry-After.

La sauvegarde n'a jamais été restaurée. Une sauvegarde prouve seulement que des octets ont été copiés. Tant qu'une restauration n'a pas été exécutée, chronométrée et vérifiée par comparaison avec les données, le délai de reprise n'est qu'une estimation ; il est souvent bien supérieur au RTO écrit. Restaurez à intervalles réguliers, dans un environnement distinct, et consignez le temps

nécessaire. Il en va de même pour la bascule : un serveur de secours qui n'a jamais pris le relais n'est qu'un espoir, et répéter la bascule dans le calme aide à découvrir les problèmes avant un incident.

Les mises à disposition provoquent les interruptions qu'elles devaient éviter. Un déploiement qui arrête l'ancienne version avant que la nouvelle soit prête laisse une interruption à chaque mise à disposition. Une modification de schéma qui réécrit une grande table sous verrou exclusif bloque toutes les requêtes la visant jusqu'à la fin. Déployez les nouvelles instances avec un contrôle vérifiant qu'elles sont prêtes, avant d'arrêter les anciennes. Répartissez les modifications de schéma entre une étape d'ajout et une étape de retrait ultérieure pour que l'ancien et le nouveau code fonctionnent entre les deux. Définissez sur Kubernetes un PodDisruptionBudget pour qu'un drainage de nœud ne puisse pas arrêter toutes les répliques en même temps.

Pour aller plus loin

- RFC 9110 : sémantique HTTP, section 15.6 — les codes d'état 5xx, dont 503 et son indication Retry-After.
- NIST SP 800-34 rév. 1 : guide de planification de secours des systèmes d'information fédéraux — les définitions de la durée maximale d'interruption tolérable, du RTO et du RPO.
- PostgreSQL : haute disponibilité, répartition de charge et réplication — serveurs de secours et bascule, décrits par le manuel de la base de données elle-même.
- PostgreSQL : archivage continu et restauration à un instant donné — la sauvegarde de base et l'archive WAL dont dépend la restauration à un instant donné.
- Kubernetes : interruptions — les interruptions volontaires et les PodDisruptionBudgets qui les limitent.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce que l'indisponibilité ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/downtime/