Engineer

Journalisation

Consignation des événements au moment où ils se produisent

Ce que c’est

La journalisation, aussi appelée journalisation des événements, consiste pour un programme à consigner son propre comportement au fil de l’eau, avec une entrée par événement : requête servie, mot de passe refusé, nouvelle tentative, arrêt brutal. Chaque entrée, ligne ou enregistrement de journal, porte au moins une heure et un message pour permettre de reconstituer ensuite ce qui s’est passé et dans quel ordre. Le projet OpenTelemetry définit un journal comme un enregistrement textuel horodaté, structuré ou non, avec des métadonnées facultatives, et note que parmi les signaux de télémétrie, les journaux sont ceux dont l’héritage historique est le plus important.

Cet héritage a sa norme. Syslog, ancien protocole d’envoi de messages d’événement à un collecteur, a été normalisé par la RFC 5424 en mars 2009, qui remplace l’ancienne RFC 3164. Chaque message reçoit l’un des huit niveaux de gravité, de 0 pour Emergency à 7 pour Debug. La plupart des bibliothèques reprennent une échelle plus courte du même principe : debug, info, warning, error. Le modèle de données des journaux d’OpenTelemetry les ramène tous à un nombre de 1 à 24, en six plages allant de TRACE à FATAL.

Une ligne non structurée est un texte libre : facile à lire pour une personne, mais une machine ne peut la décomposer qu’avec une expression régulière écrite pour sa formulation exacte. Une ligne structurée suit une structure fixe : chaque entrée possède les mêmes champs nommés, de mêmes types et de mêmes sens. OpenTelemetry préfère les journaux structurés en production et ajoute un avertissement facile à manquer : une ligne encodée en JSON n’est pas structurée simplement parce qu’elle est analysable ; c’est la stabilité de sa structure qui la rend structurée. Présentée ici sur plusieurs lignes, une entrée structurée tient sur une seule ligne sur disque :

```
{
    "time": "2026-09-14T03:12:07.412Z",
    "severity": "WARN",
    "service": "checkout",
    "event": "payment.retry",
    "attempt": 3,
    "trace_id": "4bf92f3577b34da6a3ce929d0e0e4736"
}
```

Les métriques et les traces sont les signaux habituellement collectés en parallèle. Les métriques sont des nombres échantillonnés dans le temps, peu coûteux à conserver et utiles pour dire à quelle fréquence un événement survient. Les traces suivent une requête dans tous les services traversés. Le journal garde le détail d'événements individuels : c'est ce qu'on lit après une alerte pour en comprendre la cause. OpenTelemetry relie journaux et traces en ajoutant à chaque enregistrement les identifiants de la trace et du segment actifs.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que la journalisation ? » 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

Un journal est le seul récit du passé que la plupart des systèmes laissent derrière eux. Une métrique peut dire que les erreurs ont augmenté à 03:12 ; le journal dit quelle requête, de quel service, avec quelle valeur, et ce que le programme a fait ensuite. Si une panne survient la nuit puis disparaît avant le matin, le journal est le témoin encore présent.

Ce récit se paie de trois façons. Chaque ligne consomme du disque, des entrées-sorties et du processeur, davantage encore si elle est expédiée vers un stockage central dont le service hébergé facture généralement le volume ingéré. Chaque ligne est aussi une donnée conservée sur quelqu'un : le journal d'accès d'un serveur web garde les adresses et navigateurs de tous ses visiteurs. Et chaque ligne doit être protégée. La fiche de journalisation OWASP demande que chaque

événement indique quand, où, qui et quoi, et que les journaux eux-mêmes soient protégés contre l'altération et contre l'accès, la modification et la suppression non autorisés, car ils peuvent contenir des informations personnelles ou sensibles.

À l'échelle d'une organisation, la publication spéciale NIST 800-92, le Guide de gestion des journaux de sécurité informatique de Karen Kent et Murugiah Souppaya, parue en septembre 2006, traite la gestion des journaux comme une infrastructure et des processus à l'échelle de toute l'entreprise, plutôt qu'un réglage dans un programme.

Les journaux fonctionnent aussi mieux aux côtés de nombres. Le guide d'instrumentation du projet Prometheus propose un compteur pour chaque ligne de code de journalisation : lorsqu'un message intéressant apparaît, on peut ainsi connaître sa fréquence et la période concernée. Il suggère aussi de comparer les totaux de lignes info, warning et error d'une version à la suivante.

À quoi cela ressemble en pratique

Le journal d'accès de ce site est décrit ligne par ligne sur la page de confidentialité. Chaque requête enregistre la page et l'heure, le statut de réponse, l'autodescription du navigateur, le nom du site référent et une adresse réseau partiellement masquée : sa dernière partie est remplacée par des zéros au moment d'écrire la ligne, avant toute écriture sur disque. Une ligne vit environ trente jours. Le logiciel serveur ferme chaque fichier une fois par jour, ou plus tôt vers 10 Mo, et supprime un fichier fermé vingt-huit jours plus tard ; chaque nuit, un contrôle relit tous les fichiers et échoue sur toute ligne de plus de trente jours. Deux incidents sont consignés sur la page. En septembre 2026, les adresses web peu sollicitées atteignaient si lentement 10 Mo que leurs fichiers n'avaient jamais été fermés et contenaient encore toutes les lignes depuis le début de la journalisation ; une tâche quotidienne a donc été ajoutée pour les fermer. Le 26 octobre 2026, sept des treize adresses web pour lesquelles un journal était conservé écrivaient encore dans des fichiers que cette tâche avait marqués fermés, et l'une conservait un fichier fermé vieux de quarante-sept jours. Cette tâche a été supprimée : le logiciel serveur ferme et supprime désormais lui-même les fichiers.

Une exécution complète des tests d'une application macOS a enregistré 610 996 lignes de journal du cœur logiciel. Une rafale d'événements du cœur était conservée dans un flux principal sans limite et copiée vers les flux des abonnés

dont les consommateurs ne les utilisaient pas. Le compte rendu d'incident documente séparément des rapports de diagnostic indiquant 111 Go de pages compressées sur une machine de 36 Go et un échantillon d'empreinte mémoire montrant une croissance d'environ 41 Mo par seconde lors d'une boucle de reconnexion. Des tampons bornés, des abonnements filtrés par type d'événement et un budget de débit de journalisation ont corrigé ce circuit de traitement des événements. La comparaison consignée d'exécutions complètes des tests a ramené les lignes du cœur à 1 411. Ce n'est ni un test de performance en production ni une mesure de mémoire après correction. Le budget élimine volontairement de l'information : des événements de diagnostic répétitifs peuvent être abandonnés et l'état concerné actualisé, au lieu de conserver des événements sans limite. Des délais d'arrêt distincts ont demandé leur propre correction.

Sur un hôte de production, une passe de durcissement de treize unités systemd a révélé une directive mal orthographiée dans trois modèles d'unités depuis la création de ces rôles. Face à une clé inconnue, le gestionnaire de services le signale dans le journal mais démarre tout de même l'unité : la seule trace était donc un avertissement, découvert lors de la réécriture des unités. Désormais, chaque unité passe par le vérificateur du gestionnaire de services à chaque convergence de configuration : la même faute fait donc échouer l'exécution.

Où les problèmes surviennent

Le journal grossit le plus vite quand le système est déjà en panne. Une boucle journalisant chaque nouvelle tentative écrit quelques lignes un bon jour et des milliers par seconde un mauvais jour, prenant du disque, de la mémoire et du processeur à un système sans réserve. OWASP demande de vérifier que la journalisation ne peut pas servir à épuiser les ressources, par exemple en remplissant le disque. Donnez un budget à chaque chemin très sollicité, regroupez les messages répétés en une ligne avec un nombre et gardez un compteur à côté du message pour préserver le débit mesuré lorsque des lignes sont abandonnées.

Des secrets et données personnelles sont consignés. Une requête journalisée en entier emporte tout son contenu. OWASP énumère les données généralement à supprimer, masquer, hacher ou chiffrer avant leur arrivée dans un journal : identifiants de session, jetons d'accès, mots de passe, chaînes de connexion aux bases, clés de chiffrement et données de cartes de paiement.

Faites-le à l'écriture de la ligne, comme pour les adresses dans le journal d'accès ci-dessus : au moment du nettoyage, la ligne peut déjà se trouver dans une sauvegarde ou une archive.

Une entrée extérieure fabrique une fausse ligne. Une valeur fournie par l'utilisateur contenant un retour chariot ou un saut de ligne, écrite sans échappement dans un journal textuel, peut commencer une nouvelle ligne identique à une ligne de l'application. OWASP appelle cela une injection de journal et demande d'assainir ces caractères et d'encoder les données pour leur format de destination. Un outil de journalisation structurée écrivant chaque valeur comme chaîne JSON échappée le fait par construction.

Un journal silencieux passe pour un système sain. Un disque plein, un fichier après rotation que le programme n'a jamais rouvert ou un collecteur arrêté ont le même aspect extérieur : les lignes cessent d'arriver. OWASP demande un moyen de détecter l'arrêt de la journalisation. Comptez les lignes par minute pour chaque source et déclenchez une alerte lorsque le nombre tombe à zéro : une lacune du journal devient ainsi une alerte à part entière.

Pour aller plus loin

- Journaux — OpenTelemetry sur la définition d'un journal, la préférence pour les journaux structurés et leur lien avec les traces.
- Modèle de données des journaux — les champs d'un enregistrement, ses deux horodatages et l'échelle de gravité de 1 à 24.
- RFC 5424 : protocole Syslog — le format des messages syslog et ses huit niveaux de gravité.
- NIST SP 800-92 — le guide de gestion des journaux de sécurité informatique.
- Fiche de journalisation — OWASP sur les événements à enregistrer, les données à exclure et la protection du résultat.
- Instrumentation — les pratiques du projet Prometheus pour instrumenter le code, journalisation comprise.



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 la journalisation ? » 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/logging/