



DIVERS

Log File




Introduction aux **log files**



- Les fichiers journaux enregistrent les événements et activités d'un système.
- Utilisés pour diagnostiquer, surveiller et sécuriser les systèmes informatiques.
- Présents dans tous les environnements : serveurs, applications, réseaux, systèmes d'exploitation.

Log Php sous docker



 **olymp-local-php**  f677be709c0a  [olymp-olymp-local-php:latest](#)

| Logs | Inspect | Bind mounts | Exec | Files | Stats |
|---------------------|------------|-------------|----------------------|-------|-----------------------|
| 2025-11-03 10:07:11 | 172.22.0.7 | - | 03/Nov/2025:10:07:09 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:07:13 | 172.22.0.7 | - | 03/Nov/2025:10:07:10 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:08:29 | 172.22.0.7 | - | 03/Nov/2025:10:08:27 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:08:31 | 172.22.0.7 | - | 03/Nov/2025:10:08:29 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:12:52 | 172.22.0.7 | - | 03/Nov/2025:10:12:46 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:12:59 | 172.22.0.7 | - | 03/Nov/2025:10:12:57 | +0100 | "POST /index.php" 500 |
| 2025-11-03 10:13:00 | 172.22.0.7 | - | 03/Nov/2025:10:12:59 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:13:00 | 172.22.0.7 | - | 03/Nov/2025:10:13:00 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:13:01 | 172.22.0.7 | - | 03/Nov/2025:10:12:59 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:13:55 | 172.22.0.7 | - | 03/Nov/2025:10:13:38 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:13:56 | 172.22.0.7 | - | 03/Nov/2025:10:13:55 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:21:10 | 172.22.0.7 | - | 03/Nov/2025:10:20:51 | +0100 | "POST /index.php" 200 |
| 2025-11-03 10:21:12 | 172.22.0.7 | - | 03/Nov/2025:10:21:10 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:54:02 | 172.22.0.7 | - | 03/Nov/2025:10:53:52 | +0100 | "POST /index.php" 500 |
| 2025-11-03 10:54:02 | 172.22.0.7 | - | 03/Nov/2025:10:54:02 | +0100 | "GET /index.php" 200 |
| 2025-11-03 10:54:02 | 172.22.0.7 | - | 03/Nov/2025:10:54:02 | +0100 | "GET /index.php" 200 |



Log Custom

```
[2025-10-24 13:14:01] [INFO] ✓ FLAGS vérifiés : 0/39, 0 modifications
[2025-10-24 13:14:01] [INFO] ✓ 0 mails traités pour supplier
[2025-10-24 13:14:01] [INFO] 📁 [18/18] Traitement : T&AOI-ches
[2025-10-24 13:14:02] [DEBUG] 0 messages trouvés dans T&AOI-ches
[2025-10-24 13:14:02] [DEBUG] Dossier T&AOI-ches vide, passage au suivant
[2025-10-24 13:14:02] [INFO] === Synchronisation terminée : 0 mails au total ===
[2025-10-24 13:14:02] [INFO] ✓ 0 mails récupérés du serveur IMAP
[2025-10-24 13:14:02] [DEBUG] Aucun mail à sauvegarder
[2025-10-24 13:14:02] [INFO] ✓ 0 mails enregistrés dans la base de données
[2025-10-24 13:14:02] [INFO] ✓ Synchronisation terminée pour Epicloot
[2025-10-24 13:14:03] [DEBUG] Récupération des mails pour le compte : epicloot
[2025-10-24 13:14:03] [DEBUG] Ouverture de la base de données : oxydmail.db
[2025-10-24 13:14:03] [DEBUG] Création/vérification des tables et index
[2025-10-24 13:14:03] [INFO] ✓ Base de données initialisée : oxydmail.db
[2025-10-24 13:14:04] [INFO] ✓ 966 mails récupérés pour le compte epicloot
[2025-10-24 13:14:04] [INFO] ✓ 966 mails chargés
```

Log via debugger interne (Symfony)



Symfony Exception Symfony Docs

RuntimeException > Exception HTTP 500 Internal Server Error

Neither the property "lineOne" nor one of the methods "lineOne()", "getlineOne()"/"islineOne()"/"haslineOne()" or "__call()" exist and have public access in class "stdClass" in invoicing/invoice/orderPrepare.html.twig at line 191.
Origin: https://epicloot.stagingkevinrolin.be/admin/invoicing/order
Company: EpicLoot
User: Kevinrolinbe

Exceptions 2 Logs 1 Stack Traces 2

Symfony\Component\Config\Definition\Exception
Exception

In src/EventListener/ExceptionListener.php (line 41)

```
36.         "Origin: ".$origin."\r\n".
37.         "Company: ".((($this->companyHelper->getCompany()))?$this->companyHelper->getName():'Unknown')."\r\n".
38.         "User: ".((($this->security->getUser() instanceof \Symfony\Component\Security\Core\User\UserInterface)?$this->security-
39.         ;
40.
41.         $exception = new Exception(
42.             $message,
43.             $exception->getCode(),
44.             $exception
45.         );
46.
```

in vendor/symfony/event-dispatcher/Debug/WrappedListener.php -> __invoke (line 115)

in vendor/symfony/event-dispatcher/EventDispatcher.php -> __invoke (line 206)

in vendor/symfony/event-dispatcher/EventDispatcher.php -> callListeners (line 56)



Structure d'un fichier log

Chaque entrée contient généralement :

- Une date/heure de l'événement
- Un niveau de gravité (info, warning, error)
- Une source (application, service)
- Un message descriptif

Format souvent texte brut ou JSON.



Types de fichiers journaux

- **Système** : événements du système d'exploitation.
- **Application** : comportements des logiciels.
- **Sécurité** : connexions, accès, échecs d'authentification.
- **Réseau** : trafic, erreurs, paquets rejetés.



Analyse et exploitation des logs

Permet de :

- Détecter les erreurs et incidents.
- Identifier les comportements anormaux.
- Suivre l'utilisation des ressources.

Outils : grep, awk, logrotate, Splunk, ELK (Elasticsearch, Logstash, Kibana).

Agrégateur de log (Slack)



```
# olympe-log-dev

Messages Ajouter un canevas +

"file":
"/var/www/html/olympes/src/EventListener/ExceptionListener.php:41",
"trace": [
"/var/www/html/olympes/vendor/symfony/event-
dispatcher/Debug/WrappedListener.php:115",
"/var/www/html/olympes/vendor/symfony/event-
dispatcher/EventDispatcher.php:206",
"/var/www/html/olympes/vendor/symfony/event-
dispatcher/EventDispatcher.php:56",
"/var/www/html/olympes/vendor/symfony/event-
dispatcher/Debug/TraceableEventDispatcher.php:126",
"/var/www/html/olympes/vendor/symfony/http-kernel/HttpKernel.php:241",
"/var/www/html/olympes/vendor/symfony/http-kernel/HttpKernel.php:91",
"/var/www/html/olympes/vendor/symfony/http-kernel/Kernel.php:182",

"/var/www/html/olympes/vendor/symfony/runtime/Runner/Symfony/HttpKernelRunner.
php:35",
"/var/www/html/olympes/vendor/autoload_runtime.php:29",
"/var/www/html/olympes/public/index.php:5"
],
"previous": {
"class": "Twig\\Error\\RuntimeError",
"message": "Neither the property \"lineOne\" nor one of the methods
\\\"lineOne()\\\", \\\"getLineOne()\\\", \\\"isLineOne()\\\" or \\\"hasLineOne()\\\" or
\\\"__call()\\\" exist and have public access in class \\\"stdClass\\\" in
\\\"invoicing/invoice/orderPrepare.html.twig\\\" at line 191.",
"code": 0,
"file":
"/var/www/html/olympes/templates/invoicing/invoice/orderPrepare.html.twig:191"
,
"trace": [

"/var/www/html/olympes/var/cache/dev/twig/c6/c69f44fe4d1b011b745e0fd842731385.
php:562",
"/var/www/html/olympes/vendo

OlympeLog Aujourd'hui à 9 h
```

Bonnes pratiques de gestion des logs



- Centraliser les logs.
- Mettre en place une rotation et une rétention adaptées.
- Protéger l'accès aux fichiers journaux.
- Automatiser l'analyse et les alertes.

Étude de cas / mise en pratique



Objectif : détecter une erreur dans un fichier system.log.

Étapes :

1. Ouvrir le fichier et filtrer les erreurs.
2. Identifier la source et l'heure de l'erreur.
3. Corriger le problème puis vérifier les logs après correction.

Synthèse



- Les logs sont essentiels pour la maintenance et la sécurité.
- Leur analyse aide à anticiper les pannes et incidents.
- Bien gérer les logs = meilleure fiabilité du système.