

Receiver - Les logs des modules

Sommaire

Les fichiers de log du Reactionner sont situés dans le dossier `/var/log/shinken/`. Pour plus d'informations, consultez la page [Fichiers Logs](#).

Logs spécifiques à ce démon au démarrage

Au démarrage et tous les jours à minuit, en plus des logs communs à tous les démons, le Reactionner indique l'état des loggers optionnels d'exécution :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ START-DAEMON ] [ LOGGERS CONFIGURATION ]
-----
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ START-DAEMON ] [ LOGGERS CONFIGURATION ] Optional
loggers activation :
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ START-DAEMON ] [ LOGGERS CONFIGURATION ] - DISABLED :
[ NOTIFICATION EXECUTION ]
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ START-DAEMON ] [ LOGGERS CONFIGURATION ] - DISABLED :
[ EVENTHANDLER EXECUTION ]
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ START-DAEMON ] [ LOGGERS CONFIGURATION ]
-----
```

Ces loggers peuvent être activés pour suivre l'exécution des commandes de notification et des gestionnaires d'événement (voir la section [Logs d'exécution des commandes de notification ou des gestionnaires d'événements](#)).



Si les commandes de notification ou les gestionnaires d'événement ne peuvent pas être exécutées normalement, ces loggers génèrent systématiquement des logs en erreur.

Au démarrage du démon, les lignes suivantes indiquent les limites systèmes qui sont appliquées :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ SYSTEM ] System resource number of open
files is set to (soft:1024 / hard:1024 ) (from parameter max_file_descriptor_limit)
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ SYSTEM ] System resource number of processes
/threads is set to (soft:unlimited / hard:unlimited ) (set at system max values)
```

Avec comme informations principales:

- le nombre de fichiers/socket ouvrables,
- le nombre max de processus/threads.

Gestion de la configuration

Premier chargement de la configuration

Lorsque le Reactionner reçoit sa configuration pour la première fois deux logs **INFO** sont affichés.

- Le premier indiquant que le Reactionner entre dans la phase de chargement d'une nouvelle configuration.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] ----- Loading the new
configuration from the arbiter
```

- Le deuxième indiquant que le Reactionner a reçu la configuration de l'Arbiter.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] The arbiter send us a
new configuration: [configuration_uuid=configuration-uuid, arbiter=arbiter-name,
architecture=architecture-name, date=YYYY-MM-DD HH:MM:SS]
```

- Dans le cas où le reactionner n'est pas de la même version que l'arbiter et que l'option `*mismatch_version_error*` est activé sur l'arbiter:

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ reactionner-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY]. Refusing this configuration.
```

- ⚠ Dans le cas où le reactionner n'est pas de la même version que l'arbitre et que l'option *mismatch_version_error* est désactivé sur l'arbitre:

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ reactionner-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY].
```

Mise à jour de la configuration

Lorsque qu'il y a une mise à jour de la configuration, deux logs en **INFO** sont affichés.

- Le premier indiquant que le Reactionner rentre dans la phase de chargement d'une nouvelle configuration.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] [ UPDATE ] -----  
Loading a configuration update from the arbiter
```

- Le deuxième indiquant que le Reactionner a reçu la configuration de l'Arbitre.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] [ UPDATE ] The  
arbiter send us a new configuration: [configuration_uuid=configuration-uuid, arbiter=arbiter-name,  
architecture=architecture-name, date=YYYY-MM-DD HH:MM:SS]
```

- ❌ Dans le cas où le reactionner n'est pas de la même version que l'arbitre et que l'option *mismatch_version_error* est activé sur l'arbitre:

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ reactionner-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY]. Refusing this configuration.
```

- ⚠ Dans le cas où le reactionner n'est pas de la même version que l'arbitre et que l'option *mismatch_version_error* est désactivé sur l'arbitre:

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ reactionner-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY].
```

Mise à jour des liens vers d'autres démons

Lorsque que l'Arbitre détecte un changement de lien entre les démons quatre logs en **INFO** seront affichés.

- Les deux premiers logs affichent le(les) lien(s) du(des) démon(s) supprimé(s).

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] The arbiter asked us to remove daemons:  
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] - REMOVED scheduler : [name=scheduler1-  
name] [shard_id= XXX] [uri=http://scheduler_address:port/]
```

- Les deux premiers logs affichent le(les) lien(s) du(des) démon(s) ajouté(s).

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] The arbiter send us new daemons:  
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ CONFIGURATION ] + ADDED scheduler : [name=scheduler2-  
name] [shard_id= XXX] [uri=http://scheduler_address:port/]
```

Récupération des notifications et des événements

Pour récupérer les notifications et les événements à exécuter

- **Reaktionner actif**
 - Le Reactionner va demander au Scheduler.
 - Le Reactionner indique un temps de travail disponible (en temps cpu).
 - Le Scheduler lui donne des notifications et événements (suivant le temps d'exécution moyen de ces actions constatées sur ce Reactionner). Il lui donne pour un temps inférieur ou égal au temps de travail demandé.
 - Un log permet d'avoir ce nombre de notifications et le nombre d'événements récupérés. Ce log s'affiche même si aucun événement ou notification n'a été récupéré :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ ACTIONS ] [ scheduler-master ] [ GET ] Requesting actions todo from this scheduler for 2.000s cpu time [received=11 notification(s) / 0 event(s) for 1.087s cpu time]
```

- **Reaktionner passif**
 - Le Scheduler demande au Reactionner le temps CPU disponible.
 - Le Scheduler lui envoie des notifications et événements à traiter pour le temps disponible sur le Reactionner (suivant le temps d'exécution moyen de ces actions constatées sur ce Reactionner). Il lui donne pour un temps inférieur ou égal au temps disponible sur le Reactionner.
 - Si des notifications ou événements sont reçus, un log permet d'avoir le nombre de notifications et le nombre d'événements récupérés selon le temps de travail disponible sur le Reactionner :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ ACTIONS ] [ scheduler-master ] [ RECEIVED ] We received actions todo from this scheduler for 1.000s cpu time [received=2 notification(s) / 0 event(s) for 0.160s cpu time]
```

Envoi des résultats de notifications et d'événements au Scheduler

- **Reaktionner actif**
 - Une fois les notifications et événements exécutés, le Reactionner envoie les résultats au Scheduler.
 - Un log permet d'avoir le nombre de résultats de notifications et d'événements envoyés au Scheduler et le temps mis pour être envoyé.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ ACTIONS RESULTS ] [scheduler-master] [ PUSHED ] 1 action's result(s) sends to this scheduler in [0.043]s [1 notifications / 0 events]
```

- **Reaktionner passif**
 - Une fois les notifications et événements exécutés, le Reactionner stocke les résultats en attendant que le Scheduler vienne les récupérer.
 - A chaque tour de boucle du Scheduler, ce dernier demande au Reactionner s'il a des résultats de notifications et événements disponibles.
 - Si des résultats sont disponibles, un log permet d'avoir le nombre de résultats de notifications et d'événements donné au Scheduler.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ ACTIONS RESULTS ] [scheduler-master] [ GIVEN ] 1 action's result(s) given to answer scheduler request [1 notifications / 0 events]
```

Surcharge serveur en activité disque, ralentissant l'écriture des logs

Si le serveur hébergeant le démon est surchargé en termes d'IO disques sur le volume qui héberge le fichier de log, alors ce dernier va mettre du temps à s'écrire et va ralentir tout le démon. Il faut alors si c'est faisable isoler le volume des disques sur un disque moins chargé pour ne pas ralentir le démon.

En cas de soucis, il sera écrit dans les logs :

```
YYYY-MM-DD HH:MM:SS WARNING : [ LOGGER ]
YYYY-MM-DD HH:MM:SS WARNING : [ LOGGER ]
-----
YYYY-MM-DD HH:MM:SS WARNING : [ LOGGER ] [ WRITING ] The log writes time is very high (1.87s). Please look at your log disk performance.
YYYY-MM-DD HH:MM:SS WARNING : [ LOGGER ]
-----
YYYY-MM-DD HH:MM:SS WARNING : [ LOGGER ]
```

Logs concernant les checks de vérifications Shinken

Quand un check de supervision du démon est fait, on va avoir plusieurs entrées dans les logs qui concernent des données que le démon garde sur diverses statistiques.

Un log permet d'avoir le temps pris sur le calcul des dernières commandes en timeout:

```
[YYYY-MM-DD HH:MM:SS] DEBUG : [ reactionner-name ] [ STATS ] Compute "Checks in timeouts" stats : 0.000s in a total of 2048 commands in timeouts
```

Un log permet d'avoir le temps de calcul concernant les ranges d'exécution des checks/notifications en fonction du temps (<100ms, <400ms, etc):

```
[YYYY-MM-DD HH:MM:SS] DEBUG : [ reactionner-name ] [ STATS ] Compute "Checks per CPU running time" : 0.000s (on a total of 2048 checks)
```

Un log permet d'avoir le temps de calcul pour avoir les 5 commandes les plus longues en temps CPU:

```
[YYYY-MM-DD HH:MM:SS] DEBUG : [ reactionner-name ] [ STATS ] top5 execution time 0.003s (loop over 1 ranges and 343 elements)
```

Un dernier log permet d'avoir le temps complet du calcul des statistiques du démon:

- Sur une partie commune à tous les démons (version, chemins, etc.)
- Sur la partie qui concerne uniquement ce qui concerne ce démon, donc sont inclus ici les temps précédents
- Il est affiché en :
 - **DEBUG**: si le temps de calcul est inférieur à la valeur du paramètre `display_statistics_compute_time_if_higher` du démon.
 - **INFO**: si le temps de calcul est supérieur ou égal au paramètre `display_statistics_compute_time_if_higher` du démon.

```
[YYYY-MM-DD HH:MM:SS] DEBUG : [ reactionner-name ] [ STATS ] Daemon stats were computed in 0.020s (0.001 for daemon common part, 0.020 for reactionner part)
```

En cas d'affichage INFO on met un petit morceau en plus sur comment gérer le niveau de log:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ STATS ] Daemon stats were computed in 0.004s (0.000 for daemon common part, 0.004 for poller part) (NOTE: log is displayed in INFO because 0.004 is higher than display_statistics_compute_time_if_higher=1ms in the daemon cfg)
```

Le démon nettoie ses structures de statistiques toutes les 5minutes, ce qui sera vu par la ligne de log suivante:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ STATS ] Clean checks in timeouts structure in 0.000s (before clean: 0 commands in timeouts, after clean: 0)
```

Logs de chargement des modules

Les démons ont une phase de chargement des modules qui est décrite dans la page [Reactionner - GLOBAL - Logs de gestion des modules - chapitre \[MODULES-MANAGER\]](#)

Échanges entre le processus principal et les Workers

Suivi des actions en cours

Dans le démon (boucle principale)

Le log suivant indique

- le nombre d'actions présentes dans le démon principal
- le nombre d'actions traitées sur ce tour de boucle ainsi que le nombre total d'actions présentes par Worker
- le nombre total d'actions en cours de gestion (démon + Workers)

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ STATS ] action in main daemon to be dispatched to workers: [ XX ], distribution by worker [ WORKER XX: XX done this turn / XX total pending ] [ WORKER XX: XX done this turn / XX total pending ] total: [ XX ]
```

Dans un Worker

Le log suivant indique le nombre d'actions présentes dans le Worker, ainsi qu'une estimation du temps CPU nécessaire à leur exécution

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [WORKER XX][COMMAND TO PROCESS] Worker load_todo_actions : X.XXX nb_action : XX
```

Problème temporaire de réception de résultats depuis les workers

Si un worker est surchargé, le log WARNING suivant sera généré, indiquant qu'il met trop de temps à retourner ses résultats :

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ reactionner-name ] The worker 2 reception did fail this turn (IOError), skip to the next turn to receive more.
```

- Si ces logs WARNING sont sporadiques, ils ne sont pas un problème, les résultats sont juste récupérés à la seconde suivante.
- Si ces logs WARNING sont constants, ceci signifie que le worker est saturé.

Log de performance de la boucle du Reactionner

Les logs suivants permettent de suivre le temps d'exécution de la boucle principale du Reactionner

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ REACTIONNER TIME ] [ === Loop start === ] [ Loop number=XXX ] =====  
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ REACTIONNER TIME ] [ === Loop stop === ] [ Loop number=XXX ] [PERF] [ X.XXX ]s
```

Performance des boucles des Workers

Le log suivant permet de suivre l'activité de chaque Worker et de s'assurer que chacun continue de tourner

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ REACTIONNER TIME ] [ WORKERS ] Last activity [ WORKER X: X.XXXs ago, WORKER Y: Y.YYYs ago]
```

Un Worker est lent

⚠ Quand le tour de boucle d'un Worker n'a pas fini avant un certain délai, le log suivant signale la lenteur observée

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ reactionner-name ] [ REACTIONNER TIME ] [ WORKER X ] is slow, last tick was X.XXXs ago, over limit of X.XXXs
```

Un Worker est en retard

⊗ Quand le tour de boucle d'un Worker prend beaucoup trop de temps, le log suivant signale le retard observé

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ reactionner-name ] [ REACTIONNER TIME ] [ WORKER X ] is late, last tick was X.XXXs ago, over limit of X.XXXs
```

Logs d'exécution des commandes de notification ou des gestionnaires d'événements

Gestion d'un logger

L'activation ou la désactivation des loggers optionnels se fait via l'utilisation d'une commande **curl** depuis le shell.

Les paramètres requis sont :

- REACTIONNER_IP : l'adresse IP ou le nom du serveur où tourne le Reactionner (ou *localhost* si le shell est lancé sur le serveur du Reactionner directement) ,
- REACTIONNER_PORT : le port du Reactionner,
- REACTIONNER_PROTOCOLE :
 - **https** si le protocole HTTPS a été activé sur le démon (voir le paramètre *use_ssl* dans le fichier */etc/shinken/daemons/reactionnerd.ini* du démon),
 - **http** sinon,
- **LOGGER_ID** : l'identifiant du logger, soit :
 - **NOTIFICATION_EXECUTION** pour les commandes de notification,
 - **EVENTHANDLER_EXECUTION** pour les gestionnaires d'événement.

Le résultat de cette commande renvoie un document JSON.

Activation d'un logger



Suivant l'activité du Reactionner, l'activation de ces loggers peut générer un gros volume de données.

Sur un site de production, il est conseillé de limiter l'activation de ces loggers sur de courtes périodes (*par exemple une heure au plus*), afin de limiter le surplus d'occupation disque engendré.

Commande

```
curl -s -k "REACTIONNER_PROTOCOLE://REACTIONNER_IP:REACTIONNER_PORT/set_log_enable?
logger_id=LOGGER_ID&enable=1"
```

Exemple pour activer le logger d'exécution des commandes de notification

```
curl -s -k "http://localhost:7769/set_log_enable?logger_id=NOTIFICATION_EXECUTION&enable=1"
```

La commande doit renvoyer une sortie du style :

Résultat

```
{"message": "logger [Shinken] [XXXXX EXECUTION CONTROL] is enabled"}
```

Exemple pour activer le logger d'exécution des commandes de notification

```
{"message": "logger [Shinken] [NOTIFICATION EXECUTION CONTROL] is enabled"}
```

Log d'activation d'un logger

Quand un logger est activé, le log suivant est généré :

```
[YYYY-MM-DD HH:MM:SS] INFO      : [ reactionner-name ] [ INFO-DAEMON ] [ LOGGERS CONFIGURATION ] - ENABLING  :
[ XXXX EXECUTION ]
```

```
[2025-11-06 16:45:36] INFO      : [ reactionner-master ] [ INFO-DAEMON ] [ LOGGERS CONFIGURATION ] -
ENABLING  : [ NOTIFICATION EXECUTION ]
```

Désactivation d'un logger

Commande

```
curl -s -k "REACTIONNER_PROTOCOLE://REACTIONNER_IP:REACTIONNER_PORT/set_log_enable?logger_id=LOGGER_ID&enable=0"
```

Exemple pour désactiver le logger d'exécution des commandes de notification

```
curl -s -k "http://localhost:7769/set_log_enable?logger_id=NOTIFICATION_EXECUTION&enable=0"
```

La commande doit renvoyer une sortie du style :

Résultat

```
{"message": "logger [Shinken] [XXXXX EXECUTION CONTROL] is disabled"}
```

Exemple pour désactiver le logger d'exécution des commandes de notification

```
{"message": "logger [Shinken] [NOTIFICATION EXECUTION CONTROL] is disabled"}
```

Log de désactivation d'un logger

Quand un logger est désactivé, le log suivant est généré :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ reactionner-name ] [ INFO-DAEMON ] [ LOGGERS CONFIGURATION ] - DISABLING  
: [ XXXX EXECUTION ]
```

```
[2025-11-06 16:58:23] INFO : [ reactionner-master ] [ INFO-DAEMON ] [ LOGGERS CONFIGURATION ] - DISABLING  
: [ NOTIFICATION EXECUTION ]
```

État d'un logger

Commande

```
curl -s -k "REACTIONNER_PROTOCOLE://REACTIONNER_IP:REACTIONNER_PORT/get_log_info?logger_id=LOGGER_ID"
```

Exemple pour obtenir l'état du logger d'exécution des commandes de notification

```
curl -s -k "http://localhost:7769/get_log_info?logger_id=NOTIFICATION_EXECUTION"
```

La commande doit renvoyer une sortie du style :

Résultat

```
[{"id": "NOTIFICATION_EXECUTION", "name": "[Shinken] [XXXXX EXECUTION CONTROL]", "enable": true}]
```

Exemple pour obtenir l'état du logger d'exécution des commandes de notification

```
[{"id": "NOTIFICATION_EXECUTION", "name": "[Shinken] [NOTIFICATION EXECUTION CONTROL]", "enable": false}]
```

Logs d'exécution

Exécution normale

Lorsqu'une commande s'exécute, et que le logger d'exécution a été activé, les logs suivants sont générés :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ DONE ] [ UUID ] [ COMMAND ] :  
CHEMIN_DE_LA_COMMANDE ...  
[YYYY-MM-DD HH:MM:SS] INFO : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ DONE ] [ UUID ] [ STATUS ] :  
RESULTAT D'EXÉCUTION  
[YYYY-MM-DD HH:MM:SS] INFO : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ DONE ] [ UUID ] [ RESULT ] : SORTIE  
GÉNÉRÉE LIGNE 1  
[YYYY-MM-DD HH:MM:SS] INFO : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ DONE ] [ UUID ] [ RESULT ] : SORTIE  
GÉNÉRÉE LIGNE 2
```

Exemple

```
[2025-11-06 16:52:09] INFO : [ WORKER 3 ] [ NOTIFICATION EXECUTION ] [ DONE ] [  
b526a2d704f84581a85967a02b37d9c0 ] [ COMMAND ] : /var/lib/shinken/libexec/notifications/notify_by_email.py  
...  
[2025-11-06 16:52:09] INFO : [ WORKER 3 ] [ NOTIFICATION EXECUTION ] [ DONE ] [  
b526a2d704f84581a85967a02b37d9c0 ] [ STATUS ] : 0 (OK)  
[2025-11-06 16:52:09] INFO : [ WORKER 3 ] [ NOTIFICATION EXECUTION ] [ DONE ] [  
b526a2d704f84581a85967a02b37d9c0 ] [ RESULT ] : 2025-11-06 16:52:08,858:INFO: Mail sent successfully
```

Exécution en erreur



Ces logs sont générés dans le fichier de log du Reactionner **indépendamment de l'activation** ou non du logger.

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ ERROR ] [ UUID ] [ COMMAND ] :  
CHEMIN_DE_LA_COMMANDE ...  
[YYYY-MM-DD HH:MM:SS] ERROR : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ ERROR ] [ UUID ] [ STATUS ] :  
RESULTAT D'EXÉCUTION  
[YYYY-MM-DD HH:MM:SS] ERROR : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ ERROR ] [ UUID ] [ RESULT ] :  
SORTIE GÉNÉRÉE LIGNE 1  
[YYYY-MM-DD HH:MM:SS] ERROR : [ WORKER X ] [ NOTIFICATION EXECUTION ] [ ERROR ] [ UUID ] [ RESULT ] :  
SORTIE GÉNÉRÉE LIGNE 2
```

Exemple

```
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ COMMAND ] : /usr/bin/curl ...  
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ STATUS ] : 6 (ERROR)  
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ RESULT ] : % Total % Received % Xferd Average Speed Time  
Time Time Current  
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ RESULT ] : Dload Upload Total  
Spent Left Speed  
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ RESULT ] :  
[2025-11-06 16:14:55] ERROR : [ WORKER 1 ] [ NOTIFICATION EXECUTION ] [ ERROR ] [  
af986862b07048af87a9513241182613 ] [ RESULT ] : 0 0 0 0 0 0 0 0 0 0 ---:--:-- ---:--:  
-- ---:--:-- 0curl: (6) Could not resolve host: UNSET
```