



# Configuration du serveur et de la console



From:

<https://pandorafms.com/manual/!778/>

Permanent link:

[https://pandorafms.com/manual/!778/fr/documentation/pandorafms/installation/04\\_configuration](https://pandorafms.com/manual/!778/fr/documentation/pandorafms/installation/04_configuration)

2024/12/03 19:30



# Configuration du serveur et de la console

## Introduction

Le Pandora FMS comporte trois éléments essentiels qu'il est indispensable de configurer correctement pour un **bon fonctionnement**:

- Consola web.
- PFMS server.
- Base de datos.

Cette rubrique explique les fichiers de configuration des trois éléments, ainsi que d'autres éléments importants.

## Serveur

La configuration principale du serveur FMS Pandora se trouve dans le fichier `pandora_server.conf`, dans le chemin par défaut `/etc/pandora`.

### Éléments du fichier de configuration

#### **multiprocess**

```
# If set to 1, Pandora FMS servers will run in separate processes.  
multiprocess 0
```

Désactivé par défaut. Si la valeur est fixée à 1, les serveurs Pandora FMS s'exécutent dans des processus distincts.

#### **servername**

Par défaut, il est stocké sous forme de commentaire et le nom de la machine est utilisé par le système d'exploitation.

En effet, le serveur par défaut devra être reconfiguré dans tous les agents existants pour utiliser le nouveau serveur, et l'ancien nom de serveur devra être supprimé de la liste

des serveurs.

### **incomingdir**

Répertoire d'entrée pour les paquets de données XML, par défaut in :

```
/var/spool/pandora/data_in/
```

### **log\_file**

Fichier journal (*log*), par défaut dans :

```
/var/log/pandora/pandora_server.log
```

### **snmp\_logfile**

Log de **console SNMP** par défaut dans:

```
/var/log/pandora/pandora_snmptrap.log
```

### **errorlog\_file**

Log d'erreurs, par défaut:

```
/var/log/pandora/pandora_server.error
```

### **daemon**

S'exécute en mode démon (arrière-plan), si 1 ; si 0, il s'exécute en avant-plan. Commenté par défaut. Peut également être configuré sur la ligne de commande avec l'option -D.

### **dbengine**

Base de données à utiliser, par défaut **MySQL**.

```
# dbengine: mysql  
dbengine mysql
```

**dbname**

Nom de la base de données à laquelle le serveur se connectera. Par défaut, il s'agit de `pandora`.

**dbuser**

Nom d'utilisateur pour la connexion à la base de données. La valeur par défaut est `pandora`.

**dbpass**

Mot de passe pour la connexion à la base de données de Pandora FMS.

**dbhost**

Adresse IP, URL ou nom de l'équipement qui héberge la base de données du Pandora FMS. Dans les petites installations, il s'agit généralement de l'équipement où se trouve le serveur, c'est-à-dire `127.0.0.1`.

**dbport**

Port TCP sur lequel *écoute* le moteur de base de données, par défaut 3306 est utilisé.

**verbosity**

Niveau de détail pour le serveur *logs*, de 0 (désactivé) à 10 (niveau de détail maximal).

L'utilisation continue de valeurs élevées est déconseillée en raison de la croissance importante des fichiers *log*, qui peut entraîner des problèmes de performance du système.

**master**

Priorité du serveur maître. Le serveur ayant la valeur la plus élevée (valeur numérique entière positive) est *master*. Les égalités sont résolues de manière aléatoire. S'il est fixé à 0, ce serveur ne deviendra jamais maître..

## **snmpconsole**

L'activation (avec la valeur 1) indique que la **console de réception des traps SNMP** est activée dans la configuration. Une valeur de 0 la désactive.

## **snmpconsole\_threads**

Nombre de threads de la console SNMP. Chaque thread traite un *trap* SNMP en parallèle. La valeur par défaut est 1.

## **snmpconsole\_lock**

Si la valeur est 1, les *traps* provenant de la même source ne seront jamais traités en parallèle. La valeur par défaut est 0.

## **snmpconsole\_threshold**

Temps entre deux lectures consécutives du fichier journal SNMP, en secondes. La valeur par défaut est `server_threshold`.

## **translate\_variable\_bindings**

Si la valeur est 1, la console SNMP tentera de traduire la variable `bindings` lors du traitement des traps SNMP. La valeur par défaut est 0.

## **translate\_enterprise\_strings**

Avec la valeur 1 (qui est la valeur par défaut), la console SNMP tentera de traduire les *chaînes de l'entreprise* lors du traitement des traps SNMP.

## **snmp\_ignore\_authfailure**

Le service `snmptrapd` ignore les traps SNMP de type `authenticationFailure` s'il est positionné à 1 (ce qui est le cas par défaut).

## **snmp\_pdu\_address**

Si cette option est activée (valeur 1), le service snmptrapd lira à partir de l'adresse Protocol data units (PDU) au lieu de l'adresse de l'agent. Par défaut, sa valeur est 0.

### **snmp\_trapd**

Chemin d'accès au fichier binaire snmp\_trapd. S'il est défini à manual, le serveur ne démarrera pas snmp\_trapd. La valeur par défaut est manual.

### **snmp\_forward\_trap**

Active (1) ou désactive (0) la transmission des traps SNMP à l'hôte spécifié dans le champ [snmp\\_forward\\_ip](#).

### **snmp\_forward\_ip**

Adresse IP de l'hôte vers lequel les traps SNMP seront transmis.

Veillez tout particulièrement à ne pas saisir une adresse de redirection vers le serveur Pandora FMS lui-même, car cela provoquerait une boucle de redirection et pourrait faire planter le serveur de surveillance.

### **snmp\_forward\_version**

Version SNMP à utiliser pour l'envoi de traps SNMP, l'une des valeurs suivantes: 1, 2c ó 3.

### **snmp\_forward\_secName**

Spécifie le nom de sécurité pour l'authentification [SNMP](#) version 3.

### **snmp\_forward\_engineid**

Précise l'identité autorisée ID Engine de [SNMP](#) version 3.

### **snmp\_forward\_authProtocol**

Spécifie le protocole d'authentification pour le [SNMP](#) version 3: MD5 o SHA.

### **snmp\_forward\_authPassword**

Spécifie le mot de passe d'authentification pour le **SNMP** version 3

### **snmp\_forward\_privProtocol**

Spécifie le protocole de confidentialité de **SNMP** version 3: DES o AES.

### **snmp\_forward\_privPassword**

Spécifie le mot de passe de confidentialité pour **SNMP** version 3.

### **snmp\_forward\_secLevel**

SNMP version 3 uniquement. Spécifie le niveau de sécurité. Ce paramètre ne peut prendre que les valeurs suivantes :

- noAuthNoPriv.
- authNoPriv.
- authPriv.

### **snmp\_forward\_community**

Communauté SNMP à définir (public, private, etc.).

### **networkserver**

Serveur réseau Pandora FMS : activé 1 ou désactivé 0.

### **dataserver**

Serveur de données Pandora FMS : activé 1 ou désactivé 0.

Le **Data server** est un serveur spécial qui effectue également d'autres tâches principales. Si vous avez plusieurs serveurs Pandora FMS dans votre installation, au moins l'un d'entre eux doit avoir un thread dataserver en cours d'exécution.



## **dataserver\_smart\_queue**

```
# Enable (1) or disable (0) the Data Server smart queue, which gives priority
# to new data coming from agents at the expense of buffered XML files.
dataserver_smart_queue 1
```

Lorsqu'il est activé (1), le **server** donne la priorité aux nouvelles données arrivant de chaque agent, avant les données moins récentes (mode LIFO).

## **pluginserver**

Serveur Pandora FMS remote *plugins* : activé 1 ou désactivé 0.

## **plugin\_exec**

Spécifie le chemin absolu vers le programme qui exécute *plugins* de manière contrôlée dans le temps, par défaut:

```
/usr/bin/timeout
```

Si le système de base ne dispose pas de cette commande, vous devez utiliser à la place `/usr/bin/pandora_exec`, qui est inclus dans Pandora FMS.

## **predictionserver**

Serveur de prédiction Pandora FMS : activé 1 ou désactivé 0.

## **wmiserver**

Serveur WMI Pandora FMS : activé 1 ou désactivé 0.

## **wmi\_client**

```
# WMI client binary (wmic by default).
wmi_client pandorawmic
```

Chemin complet <chemin> vers pandorawmic, par défaut:

```
/usr/bin/pandorawmic
```

## syncserver

```
# Sync Server  
#syncserver
```

Serveur de synchronisation (**Sync Server**) de Pandora FMS: activé 1 ou désactivé 0.

## network\_timeout

En secondes, délai d'attente ou *timeout* pour les vérifications ICMP. La valeur par défaut est de 2 secondes. Si vous effectuez des contrôles sur des réseaux WAN, il est conseillé d'augmenter cette valeur pour éviter les faux positifs, car certains contrôles peuvent nécessiter plus de temps.

Plus la valeur de *timeout* est élevée, plus le temps nécessaire à l'exécution des contrôles est important. Recherchez toujours une valeur étudiée et appropriée.

## server\_keepalive

En secondes, le délai avant de déclarer le serveur hors ligne. Chaque serveur vérifie l'état des serveurs qui l'entourent, et si la date de dernière mise à jour de l'un d'entre eux dépasse cette valeur, il le déclarera hors ligne. Cela affecte, dans le cas de plusieurs serveurs, le fonctionnement de **High Availability (HA)**.

Il est essentiel que, dans le cas de plusieurs serveurs, toutes leurs horloges internes soient synchronisées via NTP.

## thread\_log

La valeur par défaut est 0, sauf si le serveur Pandora FMS est en cours de débogage. La valeur 1 fait que les threads du serveur déversent périodiquement leur état sur le disque à l'emplacement suivant:

```
/tmp/<nom du serveur>.<type de serveur>.<numéro du fil>.log
```

## server\_threshold

Le nombre de secondes dans la boucle principale, en secondes. La valeur par défaut est 5.

Il s'agit d'une valeur très importante pour la configuration du serveur car elle définit le nombre de fois que Pandora FMS recherchera s'il y a des données en attente dans la base de données ou sur le disque dur (recherche de fichiers XML). Une valeur comprise entre 5 et 15 est valable dans la plupart des cas. Si la valeur est fixée à 1, la consommation de l'unité centrale augmentera considérablement. Vous pouvez utiliser la valeur 1 pour des occasions spéciales, comme lorsque, par exemple, Pandora FMS a été arrêté pendant un certain temps et qu'il y a beaucoup de fichiers XML et de tâches réseau en attente de traitement. Vous pouvez le régler sur 1, il traitera un peu plus rapidement les tâches en attente, mais lorsqu'il se termine, il devrait être réglé entre 5 et 15.

Lorsque les valeurs sont très basses et la charge élevée, un effet de "surchauffe" se produit, ce qui entraîne une augmentation progressive de la consommation du processeur et de la mémoire du serveur.

Cette valeur, ainsi que les paramètres `_thread` et `max_queue_files` du serveur, sont utilisés pour configurer les performances du serveur.

### **network\_threads**

Nombre de threads pour le serveur réseau. Indique le nombre de vérifications qui peuvent être effectuées en parallèle. Il n'est pas recommandé d'augmenter délibérément cette valeur, car cela peut entraîner une consommation excessive des ressources du serveur. Un nombre supérieur à vingt threads exige que nous disposions d'une machine avec plusieurs processeurs ou *cœurs* distincts.

### **icmp\_checks**

Définir le nombre de *pings* pour chaque module `icmp_proc`. Au moins une de ces vérifications doit retourner 1 pour que le module soit considéré comme correct. La valeur par défaut est "1". Si un nombre plus élevé est défini et que le premier *ping* réussit, les autres ne sont pas exécutés.

Dans le cas de réseaux dont la fiabilité est limitée, il est recommandé de régler 2 ou 3. Un nombre plus élevé signifie qu'en cas de défaillance d'un segment du réseau, le taux de vérification par seconde diminuera de manière significative.

A ne pas confondre avec le paramètre `icmp_packets` qui fait référence au nombre de paquets

dans l'action *ping* elle-même. La valeur `icmp_checks` définit le nombre de *pings*, chacun avec ses propres `icmp_packets`.

### **icmp\_packets**

Définit le nombre de paquets envoyés dans chaque requête ping. Valeur par défaut: 1.

### **tcp\_checks**

Nombre de tentatives TCP si la première échoue. La valeur par défaut est 1.

### **tcp\_timeout**

Délai d'attente spécifique pour les vérifications TCP. La valeur par défaut est de 30 secondes.

Un nombre élevé (supérieur à 40) entraînera une baisse significative du taux de vérification par seconde en cas de défaillance d'un segment du réseau.

### **snmp\_checks**

Nombre de tentatives SNMP si la première échoue. La valeur par défaut est 1.

### **snmp\_timeout**

Délai d'attente spécifique pour les vérifications SNMP. La valeur par défaut est de 3 secondes.

Un nombre élevé signifie qu'en cas de panne d'un segment du réseau, le taux de vérification par seconde diminuera de manière significative.

### **snmp\_proc\_deadresponse**

Retourne DOWN si un module SNMP *boolean* (type proc) ne peut pas être contacté ou s'il reçoit NULL. S'il est positionné à 0, il est ignoré.

### **plugin\_threads**

Nombre de threads pour le serveur distant *plugins*. Indique combien de vérifications peuvent être effectuées en parallèle.

### **plugin\_timeout**

Délai d'attente, en secondes, pour les vérifications à distance de *plugins*. Passé ce délai, l'état du module sera affiché comme inconnu. Sa valeur par défaut est 5, bien qu'il vaille probablement la peine de l'augmenter au cas où vous auriez des *plugins* qui prendraient plus de temps que ce délai.

### **wmi\_timeout**

Délai d'attente des vérifications WMI. Passé ce délai, l'état du module sera affiché comme inconnu. Sa valeur par défaut est de 10 secondes.

### **wmi\_threads**

Nombre de threads pour le *wmi\_server*. Indique combien de vérifications peuvent être effectuées en parallèle.

### **recon\_threads**

Nombre de threads pour le *network\_recognition\_server*. Indique combien de vérifications peuvent être effectuées en parallèle.

### **dataserver\_threads**

Nombre de threads pour le serveur de données. Ce nombre indique combien de fichiers XML peuvent être traités en parallèle. Une règle spécifique pour le serveur de données est de ne pas utiliser plus de threads que le nombre de processeurs physiques de la machine.

En fonction du nombre de XML à traiter par le serveur, une valeur normale se situe entre 1 et 4. Pour les environnements à forte charge, la valeur peut augmenter, mais le maximum est le nombre de CPU dont dispose le serveur, jamais plus.

Dans tous les cas, une valeur supérieure à 10 n'a généralement pas d'impact sur les performances, mais a un impact sur la consommation de mémoire du serveur.

## **mta\_address**

Adresse IP du serveur de messagerie (Mail Transfer Agent).

Assurez-vous que votre serveur Pandora FMS est capable de résoudre via son serveur DNS le serveur de messagerie en charge de votre domaine de messagerie.

```
nslookup -type = mx my.domain
```

Dans ce cas, assurez-vous également que votre serveur de messagerie accepte les messages redirigés depuis le serveur Pandora FMS.

Si elle n'est pas spécifiée, la [Configuration de la console Pandora FMS](#) sera utilisée. Il est possible d'avoir une configuration MTA différente pour le serveur Pandora FMS et la console Pandora FMS.

## **mta\_port**

Port du serveur de messagerie. Par défaut, il s'agit du port 25.

## **mta\_user**

Utilisateur du serveur de messagerie (si nécessaire).

## **mta\_pass**

Mot de passe du serveur de messagerie (si nécessaire).

## **mta\_auth**

Système d'authentification du serveur de messagerie, si nécessaire. Les valeurs valides sont les suivantes :

- LOGIN.
- PLAIN.
- CRAM-MD5.
- DIGEST-MD.

## mta\_from

L'adresse e-mail à partir de laquelle les e-mails seront envoyés. Par défaut, il s'agit de `pandora@localhost`.

## mta\_encryption

Type de cryptage de la connexion SMTP ( `none`, `ssl`, `starttls` ).

## mail\_in\_separate

Si la valeur est 1, la distribution du courrier sera séparée pour chaque destinataire. Si la valeur est 0, le courrier sera partagé entre tous les destinataires. Valeur par défaut 1.

## mail\_subject\_encoding

```
# Encoding to use on mail subject (MIME-Header by default)
# mail_subject_encoding MIME-Header-ISO_2022_JP
```

Désactivé par défaut. Si la langue de la console est le japonais et que le `$LANG` du serveur est `ja_jp.UTF-8`, cette clé avancée *token* permet un encodage correct des caractères dans cette langue pour les macros dans les alertes email.

## nmap

Nécessaire pour la [Discovery server](#). Par défaut, il est situé dans `/usr/bin/nmap`.

## fping

Requis pour le serveur de réseau et le serveur de réseau ICMP. Par défaut, il est situé à l'adresse `/usr/sbin/fping`.

## nmap\_timing\_template

Une valeur spécifiant la profondeur de l'analyse nmap, sur une échelle de 1 à 5. 1 signifie plus lent mais plus fiable, 5 signifie plus rapide mais moins fiable ; 2 est la valeur par défaut.

## **recon\_timing\_template**

Comme le [nmap\\_timing\\_template](#), mais appliqué aux balayages du réseau des serveurs satellites et des serveurs de découverte.

## **snmpget**

Il est nécessaire pour les contrôles SNMP. La valeur par défaut est `/usr/bin/snmpget`. Il fait référence à l'emplacement du client SNMP standard sur le système. Dans le cas de MS Windows®, un binaire est fourni à cet effet.

## **braa**

Emplacement du binaire braa, utilisé par le serveur SNMP (`/usr/bin/braa` par défaut).

## **braa\_retries**

Nombre de tentatives avant que braa ne transmette le module au serveur du réseau en cas d'erreur.

## **fsnmp**

Chemin d'accès au binaire pandorafsnmp, utilisé par le serveur SNMP pour les requêtes SNMPv3 (`/usr/bin/pandorafsnmp` par défaut).

## **autocreate\_group**

ID numérique du groupe par défaut pour les nouveaux agents créés automatiquement par la réception de fichiers de données. Si aucun groupe n'est défini ici, les agents seront créés dans le groupe contenant le XML.

## **autocreate\_group\_name**

Nom du groupe par défaut pour les nouveaux agents créés automatiquement par la réception de fichiers de données. Si aucun groupe n'est défini ici, les agents seront créés dans le groupe contenant le XML.

```
# Works like autocreate_group, except the name of the group is specified
```



```
(instead of its id).  
# Do not set both.  
#autocreate_group_name Unknown
```

Fonctionne comme `autocreate_group`, sauf que vous spécifiez le nom du groupe (au lieu de son ID). Ne pas mettre les deux.

### **autocreate\_group\_force**

Si la valeur est 1, les nouveaux agents seront ajoutés au groupe spécifié par `autocreate_group` (le groupe spécifié par l'agent sera utilisé en dernier recours).

Si la valeur est 0, les nouveaux agents sont ajoutés au groupe spécifié par l'agent (le groupe spécifié par `autocreate_group` est utilisé en dernier recours).

### **autocreate**

S'il est réglé sur 1, les agents seront créés automatiquement lors de la réception de fichiers de données avec un ID d'agent qui n'existe pas dans le système.

Si vous souhaitez mettre en place un mécanisme de sécurité, vous pouvez définir un mot de passe de groupe.

### **max\_log\_size**

Taille maximale du fichier `log` de Pandora FMS, en octets. Lorsque cette taille est atteinte, le fichier sera renommé `pandora_server.log.old` et le serveur en générera un avec le nom original, `pandora_server.log`. La taille par défaut est de 65 536 octets.

### **max\_log\_generation**

Spécifie le nombre maximum de fichiers Pandora FMS `log` ( minimum 1, maximum 9 ). La valeur par défaut est "1".

### **max\_queue\_files**

Nombre maximum de fichiers de données XML lus par le serveur de données de Pandora FMS à partir du répertoire spécifié par `incomingdir`. Cela empêche le serveur de données d'essayer de lire trop de fichiers, ce qui affecterait les performances du serveur. La valeur par défaut est 5000.

Les modules incrémentaux peuvent ne pas fonctionner correctement si cette valeur n'est pas suffisante pour contenir tous les fichiers de données XML.

### **use\_xml\_timestamp**

Par défaut, il est activé (1) et utilise la date et l'heure (*timestamp*) définies dans le XML (.data), c'est-à-dire le *timestamp* généré par l'agent.

S'il est désactivé (0), c'est le *timestamp* du fichier XML qui sera utilisé, c'est-à-dire le *timestamp* du serveur. Cela permet de désactiver globalement l'utilisation des dates générées par les agents et d'utiliser la date et l'heure du serveur comme référence pour toutes les données, puisque ce *timestamp* est généré au moment où le serveur Pandora reçoit le XML.

Cette fonction a été modifiée dans la version 747 du Pandora FMS. Dans les versions précédentes, ce *token* est désactivé par défaut.

Une fonctionnalité similaire existe au niveau de l'agent, de sorte que les données de l'agent sont évaluées avec la date de réception du fichier.

### **auto\_restart**

Il est désactivé par défaut. Si elle est activée (valeur en secondes), elle force le serveur à effectuer un redémarrage interne toutes les X secondes (1 jour = 86400). Cette option est utile si vous observez une dégradation due à la chute incontrôlée d'un thread ou d'un serveur Pandora FMS spécifique.

### **restart**

Par défaut, il est désactivé (0). En cas d'erreur critique, le serveur redémarre après un certain nombre de secondes.

Si vous utilisez `pandora_ha`, il est recommandé de mettre cette valeur à zéro et de laisser HA se charger du redémarrage si nécessaire.

### **restart\_delay**

```
# Pandora FMS will autorestart itself each XXX seconds, use this if you
experience problems with
# shutting down threads, or other stability problems.
```

```
# auto_restart 86400

# Pandora FMS will restart after restart_delay seconds on critical errors.

restart 1
restart_delay 60
```

Valeur par défaut 60. Si **restart** est défini, c'est le nombre de secondes que le serveur attendra avant de redémarrer après une erreur critique.

### **activate\_gis**

Pour activer (1) ou désactiver (0) **les fonctionnalités SIG** du serveur.

### **location\_error**

Marge ou rayon d'erreur (en mètres) permettant de considérer deux emplacements SIG comme étant le même emplacement.

### **recon\_reverse\_geolocation\_file**

Fichier contenant des informations sur la géolocalisation inversée. Ce fichier doit être au format MaxMind GPL GeoLiteCity.dat. Si cette option est commentée dans le fichier de configuration, la géolocalisation par IP sera désactivée lors de la création d'agents via *recon* et d'agents logiciels. La géolocalisation ne sera pas non plus effectuée si les fonctionnalités SIG sont généralement désactivées (**activate\_gis**).

### **recon\_location\_scatter\_radius**

Rayon (en mètres) du "cercle" à l'intérieur duquel les agents découverts par une tâche réseau seront situés. Le centre du cercle doit être calculé sur la base de la géolocalisation de l'adresse IP découverte.

### **self\_monitoring**

Le serveur dispose d'un mode d'autosurveillance qui crée un agent, portant le même nom que le serveur, qui surveille la plupart des paramètres importants d'un serveur Pandora FMS. Pour l'activer, le paramètre *self-monitoring* doit être mis à 1.

## **self\_monitoring\_interval**

Intervalle de temps, en secondes, pour **self\_monitoring**. Valeur par défaut : 300 secondes.

## **update\_parent**

Définit si l'agent peut mettre à jour son *parent* en envoyant le nom du *parent* dans le XML, mais si le paramètre n'est pas défini ou est égal à 0, les informations de l'agent seront ignorées.

Si ce n'est pas le cas, lorsque le serveur reçoit un XML avec l'attribut `parent_name`, il recherche un agent portant ce nom, et s'il le trouve, il met à jour le *parent* de l'agent dans le XML.

## **google\_maps\_description**

Cela déclenche la conversion des coordonnées GPS en une description textuelle de la position (géolocalisation inversée). L'API Google Maps sera utilisée à cet effet. Pour utiliser cette fonctionnalité, vous devez disposer d'un accès à Internet et il se peut que le traitement des informations SIG soit entravé par la vitesse de connexion à l'API de Google depuis le serveur Pandora FMS.

L'API Google Maps est un service payant qui nécessite des informations d'identification. Vous devez obtenir la CLÉ API et payer, sinon le service sera suspendu après quelques jours d'utilisation.

## **openstreetmaps\_description**

Cela déclenche la conversion des coordonnées GPS en une description textuelle de l'adresse (géolocalisation inversée). L'API **OpenStreetMaps** sera utilisée à cet effet. Ce service n'est pas aussi précis que Google Maps, mais il est gratuit. Il présente également l'avantage de pouvoir être utilisé, moyennant quelques modifications du code, pour se connecter à un serveur local.

S'il est utilisé avec une connexion Internet directe (par défaut), le traitement des informations SIG peut être moins performant en raison de la vitesse de connexion à l'API OpenStreetMaps du serveur Pandora FMS.

## **webservice**

Serveur Pandora FMS **web\_checks** : activé 1 ou désactivé 0.

### **web\_threads**

Indique combien de threads simultanés sont affectés au composant webserver.

### **web\_timeout**

Délai d'attente par défaut en secondes pour les modules de surveillance web (Goliat).

### **web\_engine**

À partir de la version 747, cURL est utilisé par défaut. Mettez LWP pour utiliser [Library for WWW in Perl \(LWP\)](#) au lieu de cURL pour la surveillance du web.

### **inventoryserver**

Serveur d'inventaire à distance Pandora FMS : activé 1 ou désactivé 0.

### **inventory\_threads**

Nombre de threads attribués au serveur d'inventaire distant.

### **exportserver**

Serveur d'exportation Pandora FMS : activé 1 ou désactivé 0.

### **export\_threads**

Nombre de threads affectés au serveur d'exportation. Indique le nombre de threads simultanés affectés à ce composant.

### **eventserver**

Alertes d'événements et serveur de corrélation : 1 activé ou 0 désactivé, anciennement appelé [correlationserver](#) .

```
# Enable (1) or disable (0) Pandora FMS Event Server.  
eventserver 0
```

## eventserver\_threads

Définit le nombre de threads à exécuter sur le **eventserver**, valeur par défaut 1.

```
# Number of threads for the Event Server.  
eventserver_threads 1
```

## event\_window

Il s'agit de la fenêtre temporelle dans laquelle le serveur de corrélation d'événements doit prendre en compte les événements.

## event\_inhibit\_alerts

Si la valeur est 1, une alerte ne sera pas exécutée (à moins qu'elle ne soit récupérée) si le dernier événement qu'elle a généré est en statut en cours. Valeur par défaut : "0".

## icmpserver

Serveur ICMP de Pandora FMS : activé 1 ou désactivé 0.

Le serveur ICMP utilise le **fping binaire** pour effectuer des requêtes ICMP en masse. Si ce composant n'est pas activé, le serveur réseau effectuera les vérifications, mais avec des performances bien moindres.

## icmp\_threads

Nombre de threads du serveur ICMP (3 par défaut).

## snmpserver

Serveur SNMP Pandora FMS : activé 1 ou désactivé 0.

Le serveur SNMP utilise le **brac|binary brac** pour exécuter des requêtes SNMP en masse. Si ce composant n'est pas activé, le serveur réseau exécutera les contrôles.

### **snmp\_threads**

Nombre de threads du serveur SNMP (3 par défaut).

### **prediction\_threads**

Nombre de threads pour le serveur de prédiction.

### **block\_size**

Taille des blocs de serveurs producteurs/consommateurs, c'est-à-dire nombre de modules par bloc (15 par défaut). Cela affecte la manière dont il traite les demandes adressées au serveur SNMP et au serveur ICMP.

### **dataserver\_lifo**

Si cette option est activée ( 1 ), les fichiers de données XML seront traités dans une pile au lieu d'une file d'attente, et les anciennes données (par exemple les données dont l'horodatage est antérieur à l'horodatage de votre module) ne déclencheront pas d'événements ou d'alertes. Désactivé, valeur ( 0 ) par défaut.

Les modules incrémentaux perdront en résolution si les fichiers de données XML s'accumulent, car les nouvelles données seront traitées en premier, ce qui entraînera l'élimination des anciennes données.

### **policy\_manager**

S'il est actif (1), le serveur écoute la file d'attente de la politique. Par défaut, sa valeur est 1.

### **event\_auto\_validation**

S'il est actif ( 1 ) les nouveaux événements créés autovalident les événements précédents du même module. Par défaut, sa valeur est 1.

### **event\_file**

Cette option de configuration vous permet de spécifier un fichier texte dans lequel les événements

générés par Pandora FMS au format CSV seront écrits. En activant cette option, une pénalité de performance de Pandora FMS est ajoutée.

Por ejemplo:

```
event_file /var/log/pandora/pandora_events.txt
```

Il n'y a pas de mécanisme de rotation pour ce fichier, vous devrez en tenir compte car il peut grandir beaucoup.

### **snmp\_storm\_protection**

```
# Set the maximum number of traps that will be processed  
# from a single source in a configured time interval.  
snmp_storm_protection 25
```

Système de protection contre les trappes SNMP par lequel la console SNMP du Pandora FMS ne traitera pas plus de ce nombre de trappes SNMP provenant d'une seule source dans un intervalle de temps défini. Si ce nombre est atteint, un événement est généré.

### **snmp\_storm\_silence\_period**

```
# Silenced time period in seconds, when trap storm is detected  
snmp_storm_silence_period 300
```

Sur détection d'une [SNMP\\_trap\\_storm](#), il entrera dans une période de silence (en secondes) définie par ce paramètre. Valeur par défaut : 300.

### **snmp\_storm\_timeout**

Intervalle de temporisation pour [snmp\\_storm\\_protection](#) en secondes.

Par exemple, pour empêcher une source unique d'envoyer plus de 1000 trappes SNMP toutes les 10 minutes:

```
snmp_storm_protection 1000  
snmp_storm_timeout 600
```

### **text\_going\_down\_normal**



Texte affiché sur les événements de module passant à l'état normal. Supporte les macros `_module_` et `_data_`.

### **text\_going\_up\_critical**

Texte affiché sur les événements de module devenant critiques. Supporte les macros `_module_` et `_data_`.

### **text\_going\_up\_warning**

Texte affiché sur les événements du module qui passent de l'état normal à l'état d'avertissement. Supporte les macros `_module_` et `_data_`.

### **text\_going\_down\_warning**

Texte affiché sur les événements de module passant de l'état critique à l'état d'avertissement. Supporte les macros `_module_` et `_data_`.

### **text\_going\_unknown**

Texte affiché sur les événements du module qui passent à un état inconnu. Supporte les macros `_module_` et `_data_`.

### **event\_expiry\_time**

Les événements plus anciens que le temps spécifié dans `event_expiry_time` (nombre de secondes) sont validés automatiquement. Pour désactiver cette fonction, réglez la valeur sur zéro (0).

### **event\_expiry\_window**

Ce paramètre est utilisé pour réduire l'impact de `event_expiry_time`, de sorte qu'il n'est pas nécessaire de vérifier l'ensemble de la table des événements. Seuls les événements plus récents que la fenêtre temporelle spécifiée (en secondes) seront validés automatiquement. Cette valeur doit être supérieure à `event_expiry_time`.

La valeur par défaut est un jour :

```
event_expiry_window 86400
```

### **claim\_back\_snmp\_modules**

Si la valeur est 1, les modules SNMP fonctionnant sur le serveur réseau seront renvoyés au serveur SNMP lorsque le script de maintenance de la base de données (pandora\_db) est exécuté.

### **async\_recovery**

S'il est réglé sur 1, les modules asynchrones qui ne reçoivent pas de données pendant deux fois leur intervalle passent à l'état normal. La valeur 0 le désactive.

### **console\_api\_url**

Adresse API de la console. Normalement, l'adresse du serveur et de la console se termine par le chemin `/include/api.php`.

### **console\_api\_pass**

Mot de passe de l'API de la console. Ce mot de passe se trouve dans la section générale de la configuration de la console et peut être vide.

### **console\_user**

Utilisateur de la console ayant les permissions d'effectuer les actions requises par l'API, telles que l'obtention d'un graphique à partir d'un module pour l'insérer dans une alerte par courrier électronique, entre autres actions.

Pour des raisons de sécurité, il est recommandé d'utiliser un utilisateur dédié à l'utilisation de l'API. Cet utilisateur ne doit pas être autorisé à accéder de manière interactive à la console, et l'utilisation de l'API doit être limitée à un ensemble d'adresses IP bien connues.

### **console\_pass**

Mot de passe de l'API **user pour la console**.

## **encryption\_passphrase**

Phrase de chiffrement utilisée pour **générer la clé du mot de passe chiffré**. Elle est commentée par défaut.

## **unknown\_events**

En cas d'activation ( 1 ) les événements des modules en statut inconnu sont activés. La valeur par défaut est 1.

## **unknown\_interval**

Intervalle de temps (en multiple de l'intervalle du module) avant que le module ne passe à l'état inconnu. Équivalent à deux fois l'intervalle de module par défaut.

## **global\_alert\_timeout**

Indique, en secondes, la durée maximale de traitement d'une alerte. Passé ce délai, l'exécution est interrompue. Par défaut, la valeur est de 15 secondes. Pour que le serveur Pandora FMS ignore ce *timeout* et ne mette jamais fin prématurément à l'exécution de l'alerte, il convient de fixer ce paramètre à 0.

## **remote\_config**

Ce paramètre contrôle s'il est possible de configurer le serveur Pandora FMS à distance depuis la console dans la vue des serveurs, 0 désactivé, 1 activé (il faut alors redémarrer le serveur PFMS). Cela fonctionne par Tentacle d'une manière similaire à la configuration à distance de **Software Agent**].

## **remote\_config\_address**

Adresse IP de la machine où vous souhaitez envoyer la configuration à distance. La valeur par défaut est localhost.

## **remote\_config\_port**

Port de **protocole Tentacle** pour la configuration à distance. Par défaut, le 41121.

## tentacle\_service\_watchdog

Version 762 ou ultérieure.

Active ou désactive la fonction **watchdog** pour les **Tentacle server**. Valeur par défaut 1 (activée), 0 pour désactiver. Voir aussi "**Démarrage et arrêt manuels des serveurs Pandora FMS**".

```
# Enable (1) or disable (0) the Tentacle Server watchdog (enabled by default).  
tentacle_service_watchdog 1
```

## remote\_config\_opts

Permet de transmettre des paramètres supplémentaires au client Tentacle pour des configurations avancées. Les paramètres doivent être placés entre guillemets (par exemple, "-v -r 5")

## warmup\_event\_interval

Spécifie le temps, en secondes, nécessaire pour générer à nouveau les événements de changement d'état et exécuter les alertes après un redémarrage du serveur.

## warmup\_unknown\_interval

Spécifie le délai, en secondes, avant que les modules puissent passer à l'état inconnu après un redémarrage du serveur.

## enc\_dir

Chemin ou *path* d'un répertoire contenant des fichiers **.enc** pour les *parser* de XML. Ces fichiers seront téléchargés par le **Data server** automatiquement.

## dynamic\_updates

Nombre de fois où les seuils dynamiques sont recalculés par plage dynamique.

## dynamic\_warning

Pourcentage par rapport à la longueur de l'intervalle critique utilisé pour le calcul des seuils de

warning. Plus il est bas, plus les intervalles de la warning et critical.

### **dynamic\_constant**

Pourcentage relatif à la moyenne d'un module utilisé pour ajuster l'écart-type d'un module lorsque les données sont constantes. Une valeur plus élevée permet d'obtenir des plages dynamiques plus larges.

### **unknown\_updates**

S'il est réglé sur "1", les modules inconnus sont vérifiés périodiquement au lieu d'être vérifiés une seule fois lorsqu'ils deviennent inconnus. Les alertes associées aux modules inconnus sont également évaluées périodiquement. 0 est la valeur par défaut.

L'utilisation de unknown\_updates avec la valeur 1 peut affecter les performances du serveur.

### **wuxserver**

Active le serveur de parsing [web user experience \(WUX\)](#). Nécessite la configuration de wux\_host et wux\_port.

### **wux\_host**

Indique l'adresse IP/le FQDN du serveur qui héberge le service Pandora Web Robot Daemon (PWRD).

### **wux\_port**

Indique le port du service Pandora Web Robot Daemon (PWRD). Sa valeur par défaut est 4444.

### **wux\_webagent\_timeout**

Temps maximum pour se connecter à une adresse web de destination et au serveur Selenium. Il est commenté par défaut, avec une valeur de 15.

## wux\_timeout

Durée maximale des transactions WUX. Valeur par défaut trente 30.

## clean\_wux\_sessions

```
# Force closing previous sessions on remote wux_host,  
# only for Selenium Grid server 3.  
#clean_wux_sessions 1
```

Si ce paramètre est activé (1) il permet de nettoyer les sessions **WUX** qui pourraient être mises en file d'attente à chaque démarrage du serveur Pandora FMS (uniquement pour Selenium 3).

## syslogserver

Serveur syslog de Pandora FMS : activé 1 ou désactivé 0.

## syslog\_file

Chemin absolu du fichier de sortie de syslog. Par exemple:

```
syslog_file /var/log/messages
```

## syslog\_threads

Nombre de threads pour le syslog.

## syslog\_max

Nombre maximum de lignes lues par le serveur syslog à chaque exécution.

## sync\_port

Port de communication du **Sync server**. Il est commenté par défaut, avec la valeur 41121.

## sync\_ca

Via le certificat de l'autorité de certification pour signer les certificats et ainsi établir une communication SSL à partir de la base de données de l'autorité de certification. [Sync server](#). Il est commenté par défaut, avec le chemin `/home/cacert.pem`.

### **sync\_cert**

Chemin d'accès au certificat du serveur pour établir une communication SSL à partir de l'ordinateur. [Sync server](#). Il est commenté par défaut, avec le chemin `/home/tentaclecert.pem`.

### **sync\_key**

Via la clé privée du certificat du serveur pour établir une communication SSI à partir du serveur [Sync server](#). Il est commenté par défaut, avec le chemin `/home/tentaclekey.pem`.

### **sync\_retries**

Nombre de tentatives de connexion au [Sync server](#). Il est commenté par défaut, avec la valeur 3.

### **sync\_timeout**

Durée maximale de la connexion avec le [Sync server](#). Il est commenté par défaut, avec la valeur 10.

### **sync\_address**

Adresse du serveur Tentacle pour le [Sync server](#).

### **ha\_interval**

Intervalle d'exécution de l'outil en secondes [Base de données HA du Pandora FMS](#). Il est commenté par défaut, avec la valeur 30.

### **ha\_monitoring\_interval**

Intervalle de surveillance en secondes de l'outil de surveillance en secondes. [Base de données HA du Pandora FMS](#). Il est commenté par défaut, avec la valeur 60.

## provisioningserver

La valeur 1 active le **Provisioning Server (Metaconsole)** de Pandora FMS, 0 le désactive.

## provisioningserver\_threads

Nombre de threads dans le **Provisioning Server (Metaconsole)** du Pandora FMS.

## provisioning\_cache\_interval

Intervalle de rafraîchissement du *cache* **Provisioning Server (Metaconsole)** de Pandora FMS en secondes (500 par défaut). Le *cache* contient tous les nœuds du Pandora FMS configurés.

## ssh\_launcher

Spécifie le chemin absolu vers le *script* `ssh_launcher.sh` qui exécute les modules d'exécution à distance. Le chemin par défaut vers *script* est:

```
/usr/share/pandora_server/util/ssh_launcher.sh
```

## rcmd\_timeout

En secondes, durée maximale d'exécution des modules d'exécution à distance. La valeur par défaut est 10.

Ce *timeout* n'a d'effet que pour indiquer le temps que le serveur Pandora FMS attendra pour obtenir des données. Les connexions seront terminées, mais cela ne garantit pas la fin de l'exécution de la commande sur la machine distante (cela doit être contrôlé par la commande elle-même).

## rcmd\_timeout\_bin

Spécifie le chemin absolu vers l'exécutable *timeout* pour les modules d'exécution à distance. Ne prend effet qu'avec l'utilisation de **ssh\_launcher**, les connexions via plink de Windows® vers Linux® et les connexions vers les systèmes MS Windows®.

- Dans Pandora FMS sur MS Windows® le chemin d'accès par défaut de l'exécutable est :

```
C:\PandoraFMS\Pandora_Server\bin\pandora_exec.exe
```



- Dans Pandora FMS sur GNU/Linux le chemin par défaut de l'exécutable est :

```
/usr/bin/timeout
```

## user y group

Vous pouvez définir dans les installations personnalisées à la fois *token* "user" et *token* "group" pour indiquer quel utilisateur et quel groupe effectueront les modifications dans les fichiers de la Console, tels que ceux liés aux politiques, aux opérations massives ou au `.conf` des agents situés à l'adresse suivante `/var/spool/pandora/data_in/conf`.

## alertserver

```
# Enable (1) or disable (0) Pandora FMS Alert Server.  
alertserver 0
```

Active ( 1 ) ou désactive ( 0 ) le serveur d'alerte. Valeur par défaut : zéro.

## alertserver\_threads

```
# Pandora FMS Alert Server threads.  
alertserver_threads 4
```

Nombre de threads à gérer par le Serveur d'alerte. Valeur par défaut : quatre.

## alertserver\_warn

```
# Generate an hourly warning event if alert execution is  
# being delayed more than alertserver_warn seconds.  
alertserver_warn 180
```

Nombre maximum de secondes pendant lesquelles l'exécution du Serveur d'alerte peut être retardée. Si cette limite est dépassée, un événement d'alerte sera généré toutes les heures. Valeur par défaut : 180 secondes.

## alertserver\_queue

```
# If set to 1, alerts are queued for the Pandora FMS Alert Server. If  
# alertserver is set to 1, alerts are always queued.  
alertserver_queue 1
```

Ce *token* permet de configurer et de mettre en file d'attente des alertes *en provenance d'autres serveurs PFMS* (nœuds) qui n'ont pas de [alertserver](#), sur un (*minimum*) ou plusieurs PFMS Alertserver(s) qui ont été installés dans l'environnement de travail.

### dbssl

```
dbssl 0
```

Active ( ' 1 ' ) ou désactive ( ' 0 ' ) l'utilisation de SSL pour la connexion à la base de données. Valeur par défaut : zéro.

### dbsslcafile

```
# dbsslcafile
```

Chemin ou emplacement du fichier, au format PEM, contenant une liste de certificats SSL émis par une autorité de certification. Il est commenté par défaut, pour l'activer, vous devez le décommenter et définir le chemin d'accès au fichier.

### dbsslcapath

```
# dbsslcapath
```

Chemin d'accès ou emplacement du répertoire ou dossier hébergeant les certificats SSL émis par une autorité de certification. Les certificats doivent être au format PEM. Il est commenté par défaut, pour l'activer, vous devez le décommenter et définir le chemin d'accès au répertoire.

### verify\_mysql\_ssl\_cert

Version 766 ou ultérieure.

```
verify_mysql_ssl_cert 0
```

Si la valeur est 1, la vérification est effectuée sur la connexion MySQL (CN du certificat SSL), *si les deux ne correspondent pas, la connexion n'est pas effectuée*. Valeur par défaut 0.

### splitbrain\_autofix

```
# Pandora FMS HA MySQL cluster splitbrain auto-recovery  
# IMPORTANT! Please understand and configure all settings from
```

```
#  
pandora_console/index.php?sec=gservers&sec2=enterprise/godmode/servers/HA_cluster&tab=setup  
# before enable this feature.  
#splitbrain_autofix 0
```

C'est un paramètre (activé avec 1 ) qui permet de récupérer automatiquement les environnements de pandora\_ha dans lesquels *Splitbrain* s'est produit, c'est-à-dire que les deux nœuds se comportent comme main ou *Master*.

Voir la section "[Haute disponibilité dans la base de données](#)" pour assurer le fonctionnement de Pandora FMS HA.

Vous devez comprendre et configurer toutes les valeurs de Servers → Manage database HA → Setup:

Pandora FMS  
the Flexible Monitoring System

Enter keywords to search

Servers / Manage Database HA  
MANAGE PANDORA DB HA

DB Replication user (i)

DB Replication user password (i)

Resync data dir (i)

Resync tmp directory (i)

Resync MySQL user (i)

Resync MySQL group (i)

Update >

Voir le [section "Récupération automatique des nœuds dans Splitbrain"](#) pour plus de détails.

### ha\_max\_splitbrain\_retries

```
# Pandora FMS HA MySQL cluster splitbrain auto-recovery settings  
# Maximum number of retries  
#ha_max_splitbrain_retries 2
```

Nombre de fois où l'auto-récupération sera effectuée en cas de défaillance la première fois que la fonction est exécutée. [Splitbrain autofix](#).

Voir le [section "Récupération automatique des nœuds dans Splitbrain"](#) pour plus de détails.

### **ha\_max\_resync\_wait\_retries**

```
# Pandora FMS HA MySQL cluster splitbrain auto-recovery settings
# Maximum number of retries to verify resync status.
#ha_max_resync_wait_retries 3
```

Nombre de fois où la synchronisation est vérifiée à la fin du processus de la fonction. [Splitbrain autofix](#).

Voir le [section "Récupération automatique des nœuds dans Splitbrain"](#) pour plus de détails.

### **ha\_resync\_sleep**

```
# Pandora FMS HA MySQL cluster splitbrain auto-recovery settings
# Maximum number of seconds waiting while verifying resync status.
#ha_resync_sleep 10
```

Secondes à écouler entre les tentatives ou *retries* configuré dans le *token anterior*; les deux paramètres appartiennent à la fonction [Splitbrain autofix](#).

Voir le [section "Récupération automatique des nœuds dans Splitbrain"](#) pour plus de détails.

### **ncmserver**

```
# Network manager configuration server.
ncmserver 1
```

Serveur NCM. Ce paramètre de configuration permet d'activer le [serveur de gestion de la configuration des périphériques réseau](#). Activé : 1, désactivé 0. La valeur par défaut est désactivée.

## ncmserver\_threads

```
# Threads for NCM server.  
ncmserver_threads 1
```

Nombre de threads dans le [serveur NCM](#).

## ncm\_ssh\_utility

```
# NCM utility to execute SSH and Telnet connections.  
ncm_ssh_utility /usr/share/pandora_server/util/ncm_ssh_extension
```

Chemin où se trouve le binaire d'exécution du [serveur NCM](#). Par défaut, il est installé dans:

```
/usr/share/pandora_server/util/ncm_ssh_extension
```

Ce binaire est utilisé pour se connecter via Telnet ou SSH aux périphériques réseau configurés dans le serveur NCM.

## correlationserver

Cette opération a été modifiée dans la version 774 du Pandora FMS : le `correlationserver` est renommé [eventserver](#).

```
# Enable (1) or disable (0) Pandora FMS Correlation Server  
correlationserver 0
```

Ce serveur *substitue* un [eventserver](#). Pour l'utiliser, il sera nécessaire de désactiver l'application `eventserver` et d'activer le `correlationserver` de cette manière:

```
eventserver 0  
correlationserver 1
```

Ce serveur évalue les alertes *corrélées* par intervalles de temps, ce qui permet d'optimiser la file d'attente dans les environnements présentant de nombreux événements simultanés.

Les méthodes `pass` et `drop` des alertes n'ont aucun effet lorsqu'elles sont activées (elles sont toujours évaluées comme `pass`). Le *pool* d'événements et *logs* est évalué à chaque seuil défini dans [correlation\\_threshold](#).

Ce serveur intègre un système de récupération des alertes *corrélées* lorsque des événements ou des *logs* cessent d'exister dans le *pool* d'évaluation qui sont conformes à l'une des règles de

l'alerte. Lorsque l'alerte est récupérée, l'action est automatiquement déclenchée avec les conditions *'retrieve'* définies dans l'action. Il n'y a pas de macros puisque le déclenchement est causé par l'absence d'informations, de sorte que la seule chose rapportée dans l'extraction est le titre de l'alerte extraite et l'heure de son extraction.

### **correlation\_threshold**

Cette opération a été modifiée dans la version 774 de Pandora FMS: le ``correlationserver`` est maintenant appelé **eventserver**.

```
# Time in seconds to re-evaluate correlation alerts pool
correlation_threshold 30
```

Temps, en secondes, pour l'évaluation du *pool* d'événements et *logs* pour le **correlationserver**.

### **preload\_windows**

```
# Pre-load windows on start with available information.
#preload_windows 0
```

Lorsque le serveur Pandora FMS est démarré, il effectue un chargement préalable des événements contenus dans la **event\_window**, pour évaluer les alertes *corrélées*. Avec le **correlationserver**, si cette option est désactivée, le redémarrage du serveur lancera une récupération pour chaque alerte qui a été lancée. Il est recommandé de l'activer afin qu'aucune récupération ne soit déclenchée à chaque redémarrage.

### **discoveryserver**

```
# Activate (1) Pandora FMS Discovery server
discoveryserver 1
```

Ce paramètre de configuration active le **serveur Discovery**. Activé : 1, désactivé 0. Par défaut, il est activé.

### **elastic\_query\_size**

```
# Log retrieving, items per request.
elastic_query_size 10
```

Éléments par requête pour la **collection\_log** (*logs*) avec Elasticsearch. Des valeurs plus élevées peuvent provoquer l'arrêt d'Elasticsearch. Valeur par défaut : ten 10.

## event\_server\_cache\_ttl

```
# Correlated Alerts, group cache ttl (in seconds). Set to 0 to disable.  
#event_server_cache_ttl 10
```

Définit, pour le [correlationserver](#), le temps de vie (en secondes) pour le groupe *cache*. Valeur par défaut lorsqu'il est activé : dix 10.

## log\_window

```
# Correlated Alerts, log window in seconds (3600 by default)  
log_window 3600
```

Définit, pour le [correlationserver](#), la période de temps (en secondes) pour la journalisation ou *log*. Valeur par défaut : 3600. Voir aussi [event\\_window](#).

## unknown\_block\_size

Version 769 ou ultérieure.

```
# Number of unknown modules that will be processed per iteration.  
unknown_block_size 1000
```

Nombre de modules inconnus à traiter dans le serveur de données PFMS, par itération (1000 par défaut).

## netflowserver

Active (1) ou désactive (0) le serveur Pandora FMS [NetFlow](#)].

```
# Enable (1) or disable (0) the Pandora FMS NetFlow Server.  
netflowserver 0
```

## netflowserver\_threads

Versión 770 o posterior.

Número de hilos para el servidor [NetFlow](#) de Pandora FMS.

```
# Number of threads for the Pandora FMS NetFlow Server.
```

```
netflowserver_threads 1
```

### syslog\_whitelist

Lors de l'activation du **Syslog server** définit les journaux autorisés par le filtrage par expressions régulières (regexp).

```
# Whitelist regexp filter for the Syslog Server.  
# syslog_whitelist .*
```

Avec `.*`, tout est permis; voir "[Filtres au niveau du serveur PFMS](#)" pour plus de détails.

### syslog\_blacklist

Lors de l'activation du **Syslog server** définit les journaux bloqués par filtrage par expressions régulières (regexp).

```
# Blacklist regexp filter for the Syslog Server.  
# syslog_blacklist regex
```

Voir "[Filtres au niveau du serveur PFMS](#)" pour plus de détails.

### critical\_on\_error

S'il est fixé à 1 (valeur par défaut), lorsque certaines vérifications échouent (pas les vérifications elles-mêmes), les modules deviennent critiques. S'il est fixé à 0, ils passeront finalement à la catégorie "inconnu".

```
critical_on_error 1
```

### logserver

Active ou désactive le serveur *logs*, valeur par défaut 0 (désactivé).

```
# Enable (1) or disable (0) Pandora FMS Log Server.  
logserver 0
```

### logserver\_threads

Définit le nombre de threads à exécuter sur le **logserver**, valeur par défaut 1.



```
# Number of threads for the Log Server.  
logserver_threads 1
```

### **too\_many\_xml**

```
# If greater than 0, generate an event when more than the specified number of  
XML data files are queued for an agent.  
too_many_xml 10
```

Valeur par défaut : 10. Génère un événement dans la console Web si le nombre de fichiers XML en attente de traitement par le serveur PFMS est supérieur au nombre spécifié.

### **agent\_deployer\_utility**

```
# Utility to deploy software agents via SSH or WinRM.  
agent_deployer_utility /usr/share/pandora_server/util/pandora_agent_deployer
```

Emplacement du programme par défaut pour le déploiement à distance des agents logiciels.

## **Variables d'environnement**

Le serveur Pandora FMS prend en charge un certain nombre d'options supplémentaires par rapport à celles proposées dans le fichier de configuration. Dans certains cas, des variables d'environnement sont nécessaires car la configuration se fait sur la machine elle-même. Pour cela, le start up du serveur *script* charge les variables d'un fichier au format BASH qui, par défaut, est:

```
/etc/pandora/pandora_server.env
```

Les variables qui peuvent être configurées sont les suivantes:

### **PANDORA\_RB\_PRODUCT\_NAME**

Cette variable est nécessaire pour personnaliser le nom du produit dans les messages initiaux affichés par le serveur. Dans le cas contraire, le nom personnalisé ne serait pas accessible tant que la base de données n'est pas chargée.

### **PANDORA\_RB\_COPYRIGHT\_NOTICE**

Cette variable est nécessaire pour personnaliser le nom de l'auteur du produit dans les messages initiaux affichés par le serveur. Sinon, le nom personnalisé ne serait pas accessible tant que la

base de données n'est pas chargée.

### Ejemplo de fichero de variables de entorno

```
#!/bin/bash
PANDORA_RB_PRODUCT_NAME="Custom product"
PANDORA_RB_COPYRIGHT_NOTICE="Custom copyright"
```

### Configuración de SNMPTRAPD

La console SNMP de Pandora FMS utilise snmptrapd pour recevoir les **SNMP traps**. Le service snmptrapd est un outil standard, présent dans presque tous les systèmes UNIX, pour recevoir les traps SNMP et écrire un fichier journal. Pandora FMS configure snmptrapd pour qu'il écrive un fichier journal personnalisé et le lise toutes les x secondes.

Auparavant, snmptrapd acceptait les traps SNMP par défaut, sans configuration explicite. À partir de la version 5.3, la configuration du contrôle d'accès est plus restrictive et ne permet pas, par défaut, de recevoir des traps SNMP de qui que ce soit.

Si snmptrapd est exécuté sans configuration personnalisée, les traps SNMP ne sont pas reçus et Pandora FMS ne peut pas les afficher dans la console, car le système les rejette.

Le plus souvent, vous devrez configurer le fichier:

```
/etc/snmp/snmptrapd.conf
```

Si le fichier ci-dessus n'existe pas, vérifiez le fichier suivant pour le débogage :

```
/var/log/pandora/pandora_snmp.log
```

Une configuration de base du fichier snmptrapd.conf serait la suivante:

```
authCommunity log public
```

Si cela ne fonctionne pas sur votre distribution Linux, veuillez vérifier la syntaxe de votre version du système snmptrapd pour permettre la réception de traps dans le *d daemon* de snmptrapd avec la commande:

```
man snmptrapd.conf
```

### Configuration des Tentacle

Vous pouvez en savoir plus sur le protocole Tentacle. [dans cette section](#).

Les Pandora FMS **Agents logiciels** envoient par défaut les paquets de données au serveur via le protocole Tentacle (port 41121/tcp attribué par IANA). Vous pouvez également reconfigurer l'agent logiciel pour qu'il envoie les données par d'autres moyens : transferts locaux (NFS, SMB) ou distants (SSH, FTP, etc.). Si vous souhaitez envoyer les paquets de données via le protocole Tentacle, vous devez configurer un serveur Tentacle pour recevoir ces données. Par défaut, lors de l'installation du serveur Pandora FMS, un serveur Tentacle est installé sur la même machine. *S'il est nécessaire d'ajuster certains paramètres de configuration du serveur Tentacle vous pouvez modifier directement le script lanceur de démon du serveur Tentacle situé à l'adresse :*

```
/etc/init.d/tentacle_serverd
```

Les différentes options de configuration du serveur Tentacle sont énumérées ci-dessous:

**PANDORA\_SERVER\_PATH**

Chemin d'accès au répertoire d'entrée des données. La valeur par défaut est:

```
/var/spool/pandora/data_in
```

**TENTACLE\_DAEMON**

Démon du Tentacle ou *daemon*. La valeur par défaut est `tentacle_server`.

**TENTACLE\_PATH**

Chemin d'accès au binaire Tentacle. Par défaut, il s'agit de:

```
/usr/bin
```

**TENTACLE\_USER**

Utilisateur avec lequel le démon Tentacle sera lancé. Par défaut, il s'agit de `pandora`.

**TENTACLE\_ADDR**

Adresse à partir de laquelle il faut écouter les paquets de données. Par défaut, il écoute dans toutes les directions, c'est-à-dire que sa valeur est `0.0.0.0`.

**TENTACLE\_PORT**

Port d'écoute pour la réception des paquets. La valeur par défaut est `41121`.

**TENTACLE\_EXT\_OPTS**

Options supplémentaires permettant d'exécuter le serveur Tentacle. Vous pouvez ici configurer Tentacle pour qu'il utilise l'authentification avec [mot de passe symétrique ou certificats](#).

#### MAX\_CONECTIONS

Nombre maximal de connexions simultanées pouvant être établies. Valeur par défaut 10.

#### MAX\_SIZE

Taille maximale du fichier pouvant être traité, en octets. Valeur par défaut 2000000.

*Voir aussi:*

- [Configuration sécurisée des Tentacle.](#)
- [Compression des données dans Tentacle.](#)

### **Pandora Web Robot Daemon (PWRD)**

Pandora Web Robot Daemon est un service qui fournit les outils nécessaires pour automatiser les sessions de navigation sur le web. Il fait partie de la fonctionnalité WUX. Il est disponible dans la bibliothèque de modules [module library](#).

Contient:

- Version binaire 46 du navigateur Mozilla Firefox®.
- Profil prédéfini pour l'enregistrement et l'exécution des sessions de navigation sur le web.
- Serveur d'automatisation des sessions.
- Enregistreur de session de navigation sur Internet ( .xpi)

Pour plus d'informations sur le PWRD, veuillez consulter le lien suivant [lien](#).

### **Configuración de multiproceso del servidor**

Version 770 ou ultérieure :

Pour les grands environnements avec plus de 50000 modules à la fois locaux (dataserver) et distants.

Si vous disposez d'une machine avec un grand nombre de cœurs et de RAM, il est pratique de séparer les processus pour les serveurs les plus exigeants (comme le Dataserver), en utilisant cette option.

Cela permettra une utilisation optimale des ressources, sans affecter les tâches du serveur principal, en déléguant la charge de travail la plus agressive à un ou plusieurs processus

secondaires sans affecter le fonctionnement du reste des composants gérés par le processus principal.

## Configuration

```
/etc/pandora/conf.d
```

Lors de la première installation de la version 770, le répertoire `conf.d` est créé, qui contiendra les fichiers pour ajouter chaque processus supplémentaire.

Le fichier `pandora_server/conf/pandora_server_sec.conf.template` doit être copié dans le répertoire `conf.d` avec une extension `.conf` avec un nom approprié (par exemple `pandora_server_sec.conf`, `pandora_server_ter.conf` et ainsi de suite).

Le fichier copié doit être édité pour respecter les règles de fonctionnement suivantes :

- Dans le fichier de configuration, il doit être défini avec un nom de serveur unique (`servername`), ne peut être la même que celle du processus principal ou d'un autre processus secondaire. Assurez-vous que `no est vide` et `no est vide` *commenté*.
- Le serveur secondaire doit toujours être `master 0`, *master* les tâches seront toujours exécutées par le serveur primaire.
- Le fichier de configuration doit avoir l'extension `.conf` et être situé dans le répertoire `conf.d`.
- Les autres configurations sont définies de la même manière que celles d'un `pandora_server standard`.
- Une fois qu'un fichier de configuration du serveur secondaire a été défini, la fonction `pandora_server` gère les serveurs primaire et secondaire, en démarrant, en arrêtant ou en signalant l'état de tous les processus configurés.

```
[root@pandorafms pandora]# /etc/init.d/pandora_server status
Pandora FMS HA is not running.
pandorafms (/etc/pandora/pandora_server.conf) Server is running with PID: 25804.
greystone3 (/etc/pandora/conf.d/pandora_server_third.conf) Server is running with PID: 25859.
greystone1 (/etc/pandora/conf.d/pandora_server_sec.conf) Server is running with PID: 25930.
[root@pandorafms pandora]# echo $?
7
[root@pandorafms pandora]# |
```

Il convient de noter que le processus de `pandora_ha` ne surveillera que le processus principal de manière dynamique et que si celui-ci se termine, pour quelque raison que ce soit, le processus `pandora_ha` redémarrera la pile entière (processus principal et secondaires).

## Console web

La [Pandora FMS web console](#) a besoin d'un serveur web pour fonctionner et utilise plusieurs langages de programmation.

## Apache web server

### Configuration d'Apache

Le Pandora FMS comporte une série de dossiers contenant des fichiers qui complètent sa fonctionnalité. Pour empêcher l'accès à ces fichiers, certains dossiers du système de gestion des droits d'auteur de Pandora ont été supprimés. `Console web` a un dossier `.htaccess` qui en limite l'accès. Pour que cette mesure soit efficace, il faut que dans la `Configuration d'Apache` doit permettre l'écrasement de ces permissions au moyen de l'option `htaccess`. C'est pourquoi vous devez définir le paramètre `token AllowOverride` sur `All`:

```
AllowOverride All
```

au lieu de:

```
AllowOverride None
```

### Fichier de configuration config.php

Les `Console web de Pandora FMS` possède un fichier de configuration qui est automatiquement généré lors de l'installation. Son emplacement est: `/consolepath/include/config.php`.

Par exemple, sur les systèmes CentOS, il se trouve à l'adresse suivante :

```
/var/www/html/pandora_console/include/config.php
```

Les options de configuration du fichier se trouvent dans l'en-tête du fichier et sont les suivantes :

```
$config["dbtype"]
```

Type de base de données utilisé. Par défaut, il s'agit de MySQL.

```
$config["dbname"]
```

Nom de la base de données de Pandora FMS. Par défaut, il s'agit de `pandora`.

```
$config["dbuser"]
```

Nom d'utilisateur pour la connexion à la base de données de Pandora FMS. Par défaut, il s'agit de `pandora`.

```
$config["dbpass"]
```

Mot de passe pour la connexion à la base de données de Pandora FMS.

```
$config["dbhost"]
```

Adresse IP ou nom de l'ordinateur où se trouve la base de données du Pandora FMS. Dans les petites installations, il s'agit généralement de l'équipement où se trouve le serveur 127.0.0.1 ou localhost.

```
$config["homedir"]
```

Répertoire dans lequel la console web du Pandora FMS est installée. Il s'agit généralement de /var/www/pandora\_console ou /srv/www/htdocs/pandora\_console.

```
$config["homeurl"]
```

Répertoire de base du Pandora FMS. Il s'agit généralement de /pandora\_console.

```
$config["public_url"]
```

Cette variable contient la valeur de l'URL interne du serveur en cas d'utilisation d'un proxy inverse tel que le mod\_proxy d'Apache.

Version 770 ou ultérieure.

```
$config["id_console"]=id;  
$config["console_description"]="description";
```

où "id" est un nombre entier supérieur à zéro.

Ces deux variables permettent de déclarer et d'ajouter des consoles pour répartir la charge d'exécution des tâches du serveur Discovery..

- Voir aussi [Discovery Console Tasks](#).
- Voir aussi [Manage Consoles](#).
- Voir aussi [Consoles de reporting dédiées](#).

## Redirection du serveur Apache

Si vous n'avez qu'une seule console Web Pandora FMS installée sur votre serveur Apache, vous pouvez vouloir rediriger automatiquement vers /pandora\_console lorsque les utilisateurs se connectent avec l'URL / du serveur Web. Pour cela, vous pouvez créer le fichier suivant index.html et le placer dans le répertoire racine du serveur web ( /var/www ou

/srv/www/htdocs ):

```
<html>
  <head>
    <meta HTTP-EQUIV="REFRESH" content="0; url=pandora_console/index.php">
  </head>
</html>
```

## Fichier de configuration php.conf

Version 768 ou ultérieure: Peut être authentifié avec API Token en envoyant le fichier [En-têtes HTTP](#) d'un *bearer token* généré par chaque utilisateur et pour son usage privé et particulier. Voir aussi "[Modifier mon utilisateur](#)".

Pour que l'authentification fonctionne correctement en utilisant *header* avec *bearer token*, il est nécessaire que la directive `HTTP_AUTHORIZATION=$1` est inclus dans le fichier `/etc/httpd/conf.d/php.conf`:

```
# Redirect to local php-fpm if mod_php (5 or 7) is not available
<IfModule !mod_php5.c>
  <IfModule !mod_php7.c>
    <IfModule !mod_php.c>
      # Enable http authorization headers
      SetEnvIfNoCase ^Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
      <Proxy "unix:/run/php-fpm/www.sock|fcgi://localhost">
        ProxySet timeout=1200
      </Proxy>

      <FilesMatch \.(php|phar)$>
        SetHandler "proxy:fcgi://localhost"
      </FilesMatch>

    </IfModule>
  </IfModule>
</IfModule>
```

[Retour à l'index de documentation du Pandora FMS](#)