

# Utilisation d'agents vRealize Log Insight

24 mai 2022

vRealize Log Insight 8.1

Vous trouverez la documentation technique la plus récente sur le site Web de VMware, à l'adresse :

<https://docs.vmware.com/fr/>

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

**VMware France SAS.**  
Tour Franklin  
100-101 Terrasse Boieldieu  
92042 Paris La Défense 8 Cedex  
France  
[www.vmware.com/fr](http://www.vmware.com/fr)

Copyright © 2022 VMware, Inc. Tous droits réservés. [Informations relatives aux copyrights et marques commerciales.](#)

# Table des matières

Utilisation d'agents vRealize Log Insight	5
<b>1</b> Présentation des agents vRealize Log Insight	6
<b>2</b> Types de schémas de rotation des journaux	9
<b>3</b> Installation ou mise à niveau des agents vRealize Log Insight	10
Télécharger les fichiers d'installation des agents	11
Installation des agents Windows	12
Installer ou mettre à jour l'agent Windows vRealize Log Insight à l'aide de l'assistant d'installation	12
Installer ou mettre à jour l'agent Windows vRealize Log Insight à partir de la ligne de commande	13
Déployer le Log Insight Windows Agent vers plusieurs machines	15
Installer ou mettre à jour le pack RPM de l'agent Linux vRealize Log Insight	19
Installer ou mettre à jour le module DEB de l'agent Linux vRealize Log Insight	21
Personnalisation de votre installation de l'agent pour Linux Debian	22
Installer le module binaire Log Insight Linux Agent	25
Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux	27
Mise à jour automatique des agents vRealize Log Insight	28
Activer ou désactiver la mise à jour automatique pour des agents spécifiques	28
<b>4</b> Configuration des agents vRealize Log Insight	30
Configurer Log Insight Windows Agent	31
Configuration par défaut de Log Insight Windows Agent	31
Collecter des événements provenant de canaux d'événements Windows	34
Collecter les événements d'un fichier journal	39
Transférer des événements vers Log Insight Windows Agent	44
Configurer Log Insight Linux Agent	44
Configuration par défaut de l'agent Linux vRealize Log Insight	45
Collecter les événements d'un fichier journal	47
Filtrage des événements provenant des agents vRealize Log Insight	56
Transfert d'informations à partir d'un agent vRealize Log Insight	57
Définir le serveur vRealize Log Insight cible	58
Spécifier la cible d'un agent	61
Configuration centralisée des agents vRealize Log Insight	65
Exemple de fusion de configurations	66
Utilisation de valeurs communes pour la configuration des agents	68

- Analyse des journaux 70
- Configurer des analyseurs de journaux 70

## 5 Désinstallation des agents vRealize Log Insight 101

- Désinstallez Log Insight Windows Agent 101
- Désinstaller le package RPM de Log Insight Linux Agent 101
- Désinstaller le module DEB de Log Insight Linux Agent 102
- Désinstaller le package bin de l'agent Linux Log Insight 102
- Désinstaller manuellement le package bin de l'agent Linux Log Insight 103

## 6 Dépannage des agents vRealize Log Insight 105

- Créer un bundle de support pour Log Insight Windows Agent 105
- Créer un bundle de support pour Log Insight Linux Agent 106
- Définir le niveau de détail des journaux dans le Log Insight Agents 106
- L'interface utilisateur d'administration n'affiche pas Log Insight Agents 107
- Les agents vRealize Log Insight n'envoient pas d'événements 108
- Ajouter une règle d'exception de trafic sortant pour Log Insight Windows Agent 109
- Autoriser les connexions sortantes à partir de Log Insight Windows Agent dans un Pare-feu Windows 111
- Le déploiement en masse de Log Insight Windows Agent a échoué 111
- Les agents Log Insight Agents rejettent les certificats auto-signés 112
- Le serveur vRealize Log Insight rejette la connexion pour trafic non chiffré 113

# Utilisation d'agents vRealize Log Insight

*Utilisation d'agents vRealize Log Insight* décrit le mode d'installation et de configuration des agents vRealize™ Log Insight™ Windows et Linux. Il inclut également des conseils de dépannage.

Ces informations sont destinées à toute personne souhaitant procéder à l'installation, à la configuration ou à la résolution des problèmes dans Log Insight Agents. Elles s'adressent aux administrateurs système Windows ou Linux expérimentés qui maîtrisent les technologies de machine virtuelle et les opérations de centre de données.

Pour obtenir des informations sur la création de classes de configuration des agents avec le serveur vRealize Log Insight, consultez *Administration de vRealize Log Insight*.

# Présentation des agents vRealize Log Insight

# 1

Un agent vRealize Log Insight collecte des événements provenant de fichiers journaux et les transfère à un serveur vRealize Log Insight ou à une destination Syslog tierce.

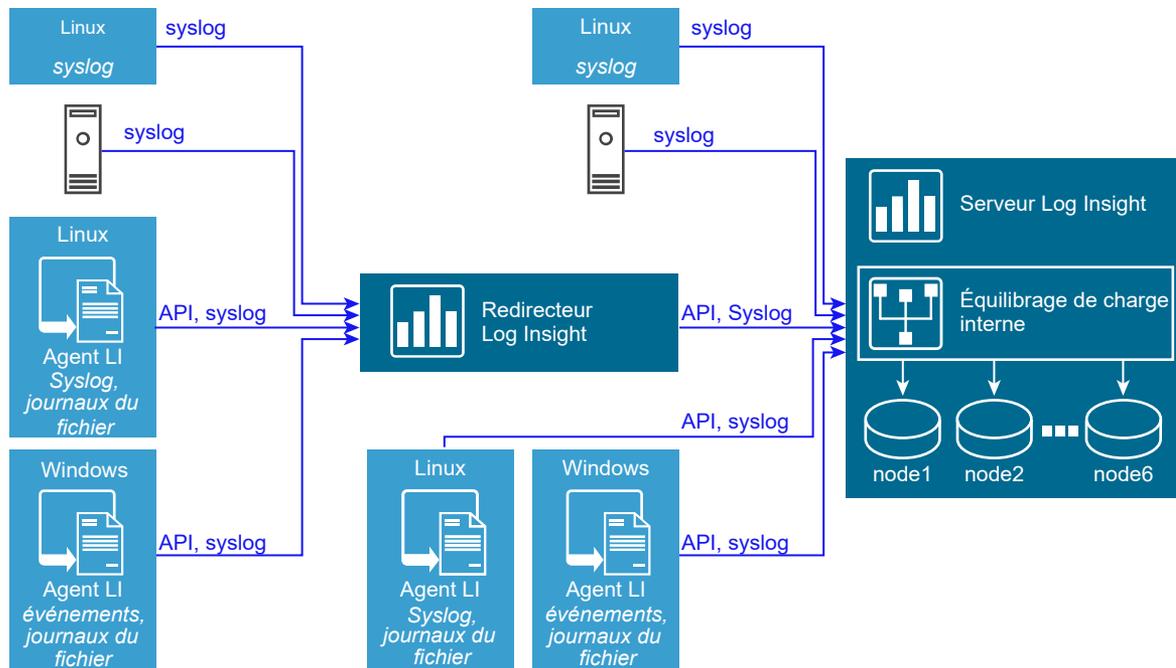
Les agents prennent en charge Syslog et l'API Ingestion vRealize Log Insight (protocole cfapi) et peuvent être utilisés avec des plates-formes Windows ou Linux. Vous configurez les agents au moyen de l'interface Web, avec le fichier liagent.ini côté serveur et client, ou dans le cadre de l'installation.

Les agents incluent les fonctionnalités suivantes :

- Déploiement simple ou de groupe
- Mise à niveau automatique
- Analyse effectuée sur les messages de journaux à des fins d'extraction des données structurées. Vous pouvez configurer des analyseurs pour les collecteurs FileLog et WinLog ou les deux.
- Prise en charge des messages multilignes
- Prise en charge native de plusieurs schémas de rotation des journaux
- Une API Ingestion complète qui inclut la compression côté client, le chiffrement et la possibilité d'ajouter des métadonnées à des événements

Le serveur vRealize Log Insight prend en charge la gestion de la configuration centralisée, ainsi que la création et la gestion des groupes d'agents.

La figure suivante représente les éléments d'une configuration de déploiement d'agent.



Un redirectionneur vRealize Log Insight est une instance dédiée d'un serveur vRealize Log Insight dont la tâche principale est de transférer des événements vers une destination distante. Normalement, une instance du serveur utilisée comme redirectionneur d'événements n'est pas utilisée pour la requête. Le redirectionneur utilise un équilibrage de charge interne et est structuré comme un serveur vRealize Log Insight.

Les agents écrivent leurs propres journaux des opérations. Pour Windows, ces journaux sont situés dans le répertoire `C:\ProgramData\VMware\Log Insight Agent\logs`. Pour Linux, le chemin d'accès au journal des opérations est `/var/log/loginsight-agent/liagent_*.log`. Les fichiers journaux sont archivés par rotation lorsqu'un agent est redémarré ou lorsque la taille du fichier atteint 10 Mo. La rotation conserve une limite combinée de 50 Mo de fichiers. Vous ne pouvez pas collecter des journaux d'agent avec l'agent vRealize Log Insight lui-même.

Les agents sont utilisés pour la collecte de journaux en temps réel. Utilisez l'importateur de vRealize Log Insight pour importer des collections de journaux historiques, y compris des bundles de support.

Des téléchargements d'installation distincts sont fournis pour les systèmes d'exploitation Windows et Linux.

Sur les systèmes Windows, l'agent s'exécute comme un service Windows et démarre immédiatement après l'installation. L'agent surveille les fichiers journaux d'application et les canaux d'événements Windows, des pools qui collectent des événements système Windows. Les événements collectés sont transmis à des serveurs vRealize Log Insight ou à des destinations Syslog tierces.

Sur les systèmes Linux, l'agent s'exécute en tant que démon et démarre immédiatement après l'installation. L'agent Linux vRealize Log Insight collecte les événements des fichiers journaux sur les machines Linux et les transfère à des serveurs vRealize Log Insight ou à des destinations Syslog. Des modules d'installation binaire Debian, Red Hat et Linux sont disponibles.

# Schémas de rotation des journaux pris en charge par les agents vRealize Log Insight

## 2

Les agents vRealize Log Insight prennent en charge plusieurs schémas de rotation des journaux.

La rotation des journaux permet de contrôler la croissance des fichiers journaux. Il existe plusieurs schémas de rotation des journaux, chacun étant conçu pour un ensemble particulier de cas d'utilisation. vRealize Log Insight inclut la prise en charge native des schémas suivants.

Tableau 2-1. Schémas de rotation des journaux pris en charge par les agents vRealize Log Insight

Schéma de rotation des journaux	Description
<code>create-new</code>	De nouveaux fichiers journaux sont créés lorsqu'une limite de taille ou de durée est atteinte. Le processus de journalisation cesse d'écrire dans le fichier journal actuel et dirige la sortie de journal vers un fichier nouvellement créé. Aucun fichier existant n'est renommé ou modifié.
<code>rename-recreate</code>	Un utilitaire externe tel que <code>logrotate</code> renomme le fichier journal lorsqu'une taille ou une limite de temps est atteinte. Le processus de journalisation crée ensuite un fichier journal avec le nom précédemment utilisé.
<code>copy-truncate</code>	Un utilitaire externe tel que <code>logrotate</code> copie le fichier journal lorsqu'une taille ou une limite de temps est atteinte. Le processus de journalisation renomme le fichier copié et réduit la taille du fichier d'origine afin que sa taille soit de 0. Le processus de journalisation peut continuer à écrire des journaux dans le fichier d'origine.

# Installation ou mise à niveau des agents vRealize Log Insight

# 3

Vous pouvez installer ou mettre à niveau les agents vRealize Log Insight sur des machines Windows ou Linux, y compris sur celles équipées de systèmes tiers de gestion des journaux.

Les agents collectent les événements et les transmettent au serveur vRealize Log Insight. Durant l'installation, vous pouvez spécifier des paramètres pour le serveur, le port et le protocole, ou conserver les valeurs par défaut.

Vous pouvez mettre à niveau les agents en utilisant les mêmes méthodes que pour l'installation ou recourir à la mise à niveau automatique. La mise à niveau automatique propage les mises à niveau aux agents lorsque vous déployez une nouvelle version de vRealize Log Insight. Pour plus d'informations, reportez-vous à [Mise à jour automatique des agents vRealize Log Insight](#). La mise à niveau n'est pas disponible pour les modules binaires Linux.

## Prise en charge du matériel

Pour installer et exécuter l'agent vRealize Log Insight, des paramètres matériels minimaux sont requis pour les hôtes/machines qui prennent en charge les architectures x86 et x86\_64, ainsi que les jeux d'instructions MMX, SSE, SSE2 et SSE3.

## Prise en charge de plate-forme

Système d'exploitation	Architecture du processeur
Windows 7, Windows 8, Windows 8.1 et Windows 10	X86_64, x86_32
Windows Server 2008, Windows Server 2008 R2,	X86_64, x86_32
Windows Server 2012, Windows Server 2012 R2, Windows Server 2016 et Windows Server 2019	x86_64
RHEL 5, RHEL 6, RHEL 7 et RHEL 8	X86_64, x86_32
SuSE Linux Entreprise (SLES) 11 SP3 et SLES 12 SP1	x86_64
Ubuntu 14.04 LTS, Ubuntu 16.04 LTS et Ubuntu 18.04	x86_64
VMware Photon, version 1 révision 2, version 2 et version 3	x86_64

## Notes relatives à Linux

Si vous implémentez une installation par défaut de l'agent Linux Log Insight pour un utilisateur dépourvu de privilèges racine, la configuration par défaut peut provoquer des problèmes liés à la collecte des données. L'agent ne consigne pas un avertissement indiquant que l'abonnement au canal a échoué, et les fichiers de la collecte ne disposent pas des autorisations de lecture. Le message `Inaccessible log file ... will try later` est ajouté à plusieurs reprises au journal. Vous pouvez commenter la configuration par défaut source du problème ou vous pouvez modifier les autorisations utilisateur.

Si vous installez les agents Linux à l'aide d'un package rpm ou DEB, le script `init.d` nommé `liagentd` est installé dans le cadre de l'installation du package. Le package bin ajoute le script mais ne l'enregistre pas. Vous pouvez enregistrer le script manuellement.

Pour vérifier que l'installation s'est bien déroulée, exécutez la commande `status liagentd` du service (`/sbin/`).

Ce chapitre contient les rubriques suivantes :

- [Télécharger les fichiers d'installation des agents](#)
- [Installation des agents Windows](#)
- [Installer ou mettre à jour le pack RPM de l'agent Linux vRealize Log Insight](#)
- [Installer ou mettre à jour le module DEB de l'agent Linux vRealize Log Insight](#)
- [Personnalisation de votre installation de l'agent pour Linux Debian](#)
- [Installer le module binaire Log Insight Linux Agent](#)
- [Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux](#)
- [Mise à jour automatique des agents vRealize Log Insight](#)

## Télécharger les fichiers d'installation des agents

La première étape de configuration d'un agent vRealize Log Insight consiste à télécharger un module d'installation de l'agent pour votre plate-forme.

Le nom d'hôte de destination est ajouté à tous les modules téléchargés à partir de la page **Agents** du serveur vRealize Log Insight. `server.hostname` est appliqué lors d'une installation initiale pour les agents MSI, RPM et DEB. Si un nom d'hôte est déjà présent dans le fichier de configuration ou si vous exécutez le module avec le paramètre `hostname`, le nom d'hôte du serveur intégré est ignoré.

### Procédure

- 1 Accédez à l'onglet **Administration** de l'interface utilisateur Web de vRealize Log Insight.
- 2 Dans la section **Gestion**, cliquez sur **Agents**.
- 3 Accédez au bas de l'écran, puis cliquez sur **Télécharger l'agent Log Insight**.

- 4 Pour télécharger un module d'installation, sélectionnez-le dans le menu contextuel, puis cliquez sur **Enregistrer**.

Option	Description
MSI Windows	Module d'installation pour plate-forme Windows (32 bits/64 bits)
RPM Linux	Module d'installation pour Linux Red Hat, openSuSE (32 bits/64 bits) ou VMware Photon Platform
DEB Linux	Module d'installation pour plate-forme Linux Debian (32 bits/64 bits)
BIN Linux	Module d'installation automatique pour Linux (32 bits/64 bits). Il n'est pas nécessaire d'utiliser un système de gestion des modules.

### Étape suivante

Utilisez les fichiers téléchargés pour déployer l'agent vRealize Log Insight.

## Installation des agents Windows

Vous pouvez installer un agent sur une machine spécifique à l'aide de l'assistant d'installation ou de la ligne de commande, ou déployer plusieurs instances d'un agent à l'aide d'un script.

### Mise à niveau des agents Windows

Vous pouvez mettre à niveau un agent Windows en appliquant un fichier de mise à niveau à l'aide de l'une des méthodes utilisées pour l'installation. Vous pouvez également utiliser la fonctionnalité de mise à niveau automatique pour mettre à niveau vos agents en arrière-plan.

## Installer ou mettre à jour l'agent Windows vRealize Log Insight à l'aide de l'assistant d'installation

Vous pouvez installer ou mettre à niveau un agent Windows sur une machine spécifique à l'aide de l'assistant d'installation.

### Conditions préalables

- Vérifiez que vous disposez d'une copie du fichier `.msi` de l'agent vRealize Log Insight Windows. Reportez-vous à [Télécharger les fichiers d'installation des agents](#).
- Vérifiez que vous disposez des autorisations pour effectuer des installations et démarrer des services sur la machine Windows.

### Procédure

- 1 Connectez-vous à la machine Windows sur laquelle vous souhaitez installer l'agent vRealize Log Insight Windows.
- 2 Accédez au répertoire dans lequel se trouve le fichier `.msi` de l'agent vRealize Log Insight Windows.

- 3 Double-cliquez sur le fichier `.msi` de l'agent Windows vRealize Log Insight, acceptez les conditions du contrat de licence et cliquez sur **Suivant**.
- 4 Entrez l'adresse IP ou le nom d'hôte du serveur vRealize Log Insight, puis cliquez sur **Installer**.  
L'assistant installe ou met à jour l'agent Windows vRealize Log Insight en tant que service Windows automatique sous le compte de service Local System.
- 5 Cliquez sur **Terminer** .

#### Étape suivante

Pour configurer l'agent vRealize Log Insight Windows, modifiez le fichier `liagent.ini`. Voir [Configurer Log Insight Windows Agent](#).

## Installer ou mettre à jour l'agent Windows vRealize Log Insight à partir de la ligne de commande

Vous pouvez installer ou mettre à jour l'agent Windows à partir de la ligne de commande.

Vous pouvez utiliser le compte de service par défaut ou en spécifier un autre, et utiliser les paramètres de la ligne de commande pour indiquer les informations relatives au serveur, au port et au protocole. Pour des options de ligne de commande MSI, accédez au site Web de la bibliothèque Microsoft Developer Network (MSDN) et recherchez-y les options de ligne de commande MSI en question.

#### Conditions préalables

- Vérifiez que vous disposez d'une copie du fichier `.msi` de l'agent vRealize Log Insight Windows. Reportez-vous à [Télécharger les fichiers d'installation des agents](#).
- Vérifiez que vous disposez des autorisations pour effectuer des installations et démarrer des services sur la machine Windows.
- Si vous utilisez une option d'installation silencieuse (`/quiet` ou `/qn`), assurez-vous d'exécuter l'installation en tant qu'administrateur. Si vous n'êtes pas un administrateur et que vous exécutez une installation silencieuse, vous n'êtes pas invité à entrer les privilèges d'administrateur et l'installation échoue. Pour les diagnostics, utilisez l'option de connexion et les paramètres `/!xv* file_name`.

#### Procédure

- 1 Connectez-vous à la machine Windows sur laquelle vous souhaitez installer ou mettre à jour l'agent vRealize Log Insight Windows.
- 2 Ouvrez une fenêtre d'**invite de commande**.
- 3 Accédez au répertoire dans lequel se trouve le fichier `.msi` de l'agent vRealize Log Insight Windows.

- 4 Effectuez l'installation ou la mise à jour en utilisant les valeurs par défaut et une commande au format suivant. Remplacez la séquence *version-build\_number* par votre numéro de version et de build.

L'option `/quiet` permet d'exécuter la commande en mode silencieux et l'option `/lxv` crée un fichier journal dans le répertoire actif.

```
Drive:\path-to-msi_file>VMware-Log-Insight-Agent-version-build_number.msi /
quiet /lxv* li_install.log
```

- 5 (Facultatif) Spécifiez un compte de service d'utilisateur sous lequel exécuter le service de l'agent Windows vRealize Log Insight.

```
Drive:\path-to-msi_file>VMware-Log-Insight-Agent-*.msi SERVICEACCOUNT=domain\user
SERVICEPASSWORD=user_password
```

**Note** Le compte fourni dans le paramètre `SERVICEACCOUNT` obtient le droit **Ouvrir une session en tant que service** et l'accès complet en écriture au répertoire `%ProgramData%\VMware\Log Insight Agent`. Ce compte est créé s'il n'existe pas encore. Le nom d'utilisateur ne doit pas dépasser 20 caractères. Si vous ne spécifiez pas de paramètre `SERVICEACCOUNT`, le service de l'agent Windows vRealize Log Insight est installé ou mis à jour sous le compte de service `LocalSystem`.

- 6 (Facultatif) Vous pouvez spécifier des valeurs pour les options de ligne de commande suivantes selon vos besoins.

Option	Description
<code>SERVERHOST=hostname</code>	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight.
<code>SERVERPROTO=protocol</code>	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .
<code>SERVERPORT=portnumber</code>	Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut. <ul style="list-style-type: none"> <li>■ <code>cfapi</code> avec SSL activé : 9543</li> <li>■ <code>cfapi</code> avec SSL désactivé : 9000</li> <li>■ <code>syslog</code> avec SSL activé : 6514</li> <li>■ <code>syslog</code> avec SSL désactivé : 514</li> </ul>

Option	Description
<b>SERVICEACCOUNT=nom-de-compte</b>	Compte de service de l'utilisateur sous lequel le service de Log Insight Windows Agentest exécuté.  <b>Note</b> Le compte fourni dans le paramètre SERVICEACCOUNT doit disposer du privilège <b>Se connecter en tant que service</b> et d'un accès en écriture au répertoire %ProgramData%\VMware\Log Insight Agent pour que le programme d'installation s'exécute correctement. Si vous ne spécifiez pas de paramètre SERVICEACCOUNT, le service de l'agent vRealize Log Insight Windows est installé sous le compte de service LocalSystem.
<b>SERVICEPASSWORD=Mot mot de passe</b>	Mot de passe du compte de service de l'utilisateur.
<b>AUTOUPDATE={yes no}</b>	Active ou désactive la mise à jour automatique de l'agent. Vous devez également activer la mise à jour automatique à partir du serveur vRealize Log Insight pour activer la mise à jour automatique. La valeur par défaut est Yes.
<b>LIAGENT_SSL={yes no}</b>	Activez une connexion sécurisée. Si SSL est activé, l'agent utilise le protocole TLS 1.2 pour communiquer avec le serveur. La valeur par défaut est Yes.

## Résultats

La commande installe ou met à jour l'agent Windows vRealize Log Insight en tant que service Windows. Le service de l'agent Windows vRealize Log Insight démarre lorsque vous démarrez la machine Windows.

## Étape suivante

Vérifiez que les paramètres de ligne de commande que vous définissez sont correctement appliqués dans le fichier `liagent.ini`. Voir [Configurer Log Insight Windows Agent](#).

## Déployer le Log Insight Windows Agent vers plusieurs machines

Vous pouvez réaliser un déploiement en masse de Log Insight Windows Agent sur des ordinateurs cibles dans un domaine Windows.

### Procédure

#### 1 Créer un fichier de transformation pour déployer plusieurs agents Windows vRealize Log Insight

Dans le cadre du déploiement de plusieurs agents, vous devez créer un fichier de transformation qui spécifie les paramètres de configuration du déploiement. Le fichier de transformation `.mst` est appliqué au fichier `.msi` lorsque vous installez des agents. Les paramètres incluent le serveur de destination pour les agents, ainsi que le protocole de communication, un port et un compte d'utilisateur pour installer et démarrer le service d'agent Log Insight.

#### 2 Déployer plusieurs instances de l'agent Windows vRealize Log Insight

Vous pouvez déployer plusieurs instances de l'agent Windows vRealize Log Insight sur des ordinateurs cibles dans un domaine Windows.

## Créer un fichier de transformation pour déployer plusieurs agents Windows vRealize Log Insight

Dans le cadre du déploiement de plusieurs agents, vous devez créer un fichier de transformation qui spécifie les paramètres de configuration du déploiement. Le fichier de transformation .mst est appliqué au fichier .msi lorsque vous installez des agents. Les paramètres incluent le serveur de destination pour les agents, ainsi que le protocole de communication, un port et un compte d'utilisateur pour installer et démarrer le service d'agent Log Insight.

Les paramètres incluent le serveur de destination pour les agents, ainsi que le protocole de communication, un port et un compte d'utilisateur pour installer et démarrer le service d'agent Log Insight.

### Conditions préalables

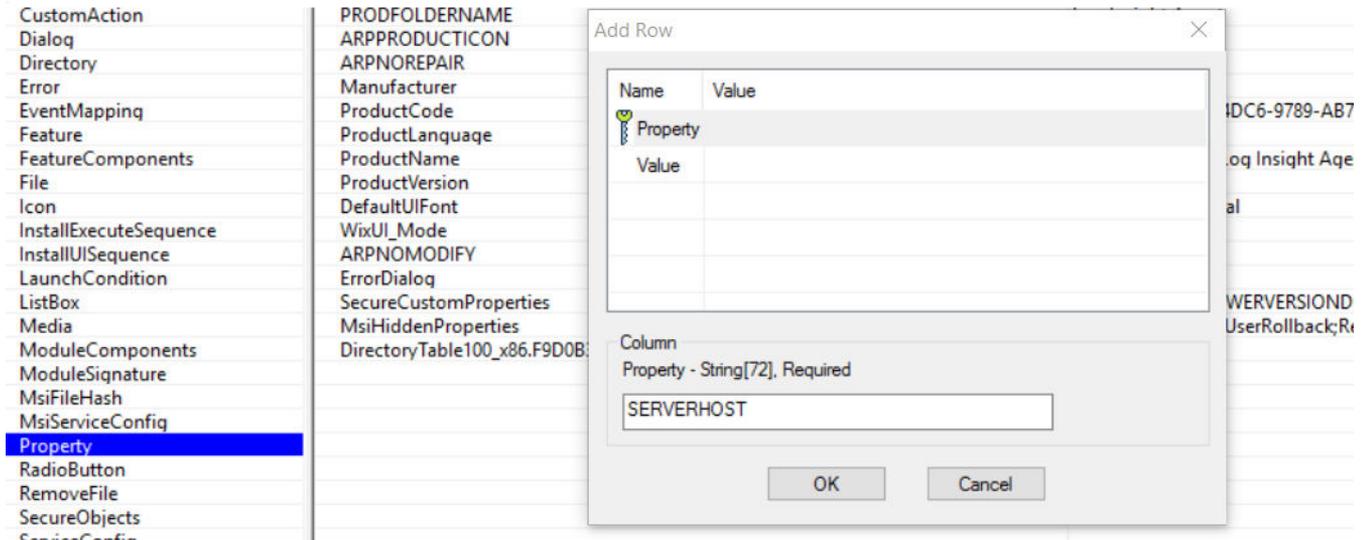
- Vérifiez que vous disposez d'une copie du fichier .msi vRealize Log Insight Windows. Reportez-vous à [Télécharger les fichiers d'installation des agents](#).
- Téléchargez et installez l'éditeur de base de données Orca. Voir <http://support.microsoft.com/kb/255905>.

### Procédure

- 1 Ouvrez le fichier .msi de l'agent Windows vRealize Log Insight dans l'éditeur Orca et sélectionnez **Transformation > Nouvelle transformation**.

- 2 Modifiez la table de propriétés et ajoutez les paramètres et les valeurs nécessaires pour une installation ou une mise à niveau personnalisée.

Figure 3-1. Table de propriétés



- a Cliquez sur **Propriété**.
- b Dans le menu déroulant **Table**, sélectionnez **Ajouter une ligne**.
- c Entrez le nom d'une propriété et une valeur dans la boîte de dialogue Ajouter une ligne.

Les paramètres sont affichés dans le tableau suivant.

Paramètre	Description
<b>SERVERHOST</b>	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. La valeur par défaut est <b>loginsight</b> .
<b>SERVERPROTO</b>	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <b>cfapi</b> et <b>syslog</b> . La valeur par défaut est <b>cfapi</b> .
<b>SERVERPORT</b>	Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut. <ul style="list-style-type: none"> <li>■ cfapi avec SSL activé : 9543</li> <li>■ cfapi avec SSL désactivé : 9000</li> <li>■ syslog avec SSL activé : 6514</li> <li>■ syslog avec SSL désactivé : 514</li> </ul>

Paramètre	Description
<b>SERVICEACCOUNT</b>	Compte de service de l'utilisateur sous lequel le service de Log Insight Windows Agentest exécuté.  <b>Note</b> Le compte fourni dans le paramètre <code>SERVICEACCOUNT</code> doit disposer du privilège <b>Se connecter en tant que service</b> et d'un accès en écriture au répertoire <code>%ProgramData%\VMware\Log Insight Agent</code> pour que le programme d'installation s'exécute correctement. Si vous ne spécifiez pas de paramètre <code>SERVICEACCOUNT</code> , le service de l'agent vRealize Log Insight Windows est installé sous le compte de service LocalSystem.
<b>SERVICEPASSWORD</b>	Mot de passe du compte de service de l'utilisateur.
<b>AUTOUPDATE</b>	Active ou désactive la mise à jour automatique de l'agent. Vous devez également activer la mise à jour automatique à partir du serveur vRealize Log Insight pour activer la mise à jour automatique. La valeur par défaut est Yes.
<b>LIAGENT_SSL</b>	Activez une connexion sécurisée. Si SSL est activé, l'agent utilise le protocole TLS 1.2 pour communiquer avec le serveur. La valeur par défaut est Yes.

- 3 Sélectionnez **Transformation > Générer une transformation** et enregistrez le fichier de transformation `.mst`.

#### Étape suivante

Utilisez les fichiers `.msi` et `.mst` pour déployer l'agent Windows vRealize Log Insight.

## Déployer plusieurs instances de l'agent Windows vRealize Log Insight

Vous pouvez déployer plusieurs instances de l'agent Windows vRealize Log Insight sur des ordinateurs cibles dans un domaine Windows.

Pour savoir pourquoi il peut être nécessaire de redémarrer la machine du client deux fois, consultez l'article <http://support.microsoft.com/kb/305293>.

#### Conditions préalables

- Vérifiez que vous disposez d'un compte administrateur ou d'un compte avec des privilèges d'administrateur sur le contrôleur de domaine.
- Vérifiez que vous disposez d'une copie du fichier `.msi` de l'agent vRealize Log Insight Windows. Reportez-vous à [Télécharger les fichiers d'installation des agents](#).
- Familiarisez-vous avec les procédures décrites dans les articles <http://support.microsoft.com/kb/887405> et <http://support.microsoft.com/kb/816102>.

#### Procédure

- 1 Connectez-vous au contrôleur de domaine en tant qu'administrateur ou utilisateur disposant de privilèges d'administration.
- 2 Créez un point de distribution et copiez le fichier `.msi` de l'agent Windows vRealize Log Insight sur le point de distribution.

- 3 Ouvrez la console de gestion de la stratégie de groupe et créez un objet de stratégie de groupe pour déployer le fichier `.msi` de l'agent Windows vRealize Log Insight.
- 4 Modifiez l'objet de stratégie de groupe pour le déploiement du logiciel et attribuez un package.
- 5 (Facultatif) Si vous avez généré un fichier `.mst` avant le déploiement, sélectionnez le fichier de configuration `.mst` dans l'onglet **Modifications** de la fenêtre **Propriétés du GPO**. Ensuite, modifiez un objet de stratégie de groupe selon la méthode Avancée pour déployer le package `.msi`.
- 6 (Facultatif) Mettez à niveau l'agent Windows vRealize Log Insight.
  - a Copiez le fichier `.msi` de la mise à niveau vers le point de distribution.
  - b Cliquez sur l'onglet **Mise à niveau** de la fenêtre **Propriétés** de l'objet de stratégie de groupe.
  - c Ajoutez la version installée initialement du fichier `.msi` dans la section Packages qui seront mis à niveau par ce package.
- 7 Déployez l'agent Windows vRealize Log Insight vers des groupes de sécurité spécifiques qui incluent les utilisateurs de domaine.
- 8 Fermez toutes les fenêtres de la console de gestion de la stratégie de groupe et de l'éditeur de gestion de la stratégie de groupe sur le contrôleur de domaine, puis redémarrez les machines du client.

Si l'option Optimisation de connexion rapide est activée, redémarrez deux fois la machine du client.
- 9 Vérifiez que l'agent Windows vRealize Log Insight est installé sur les machines du client en tant que service local.

Si vous avez configuré les paramètres `SERVICEACCOUNT` et `SERVICEPASSWORD` pour l'utilisation d'un fichier `.mst` afin de déployer plusieurs instances de l'agent Windows vRealize Log Insight, vérifiez que l'agent Windows vRealize Log Insight est installé sur les machines du client, sous le compte d'utilisateur que vous avez spécifié.

#### Étape suivante

Si les différentes instances de l'agent Windows vRealize Log Insight ne sont pas fonctionnelles, consultez la section [Le déploiement en masse de Log Insight Windows Agent a échoué](#).

## Installer ou mettre à jour le pack RPM de l'agent Linux vRealize Log Insight

Vous pouvez installer ou mettre à jour l'agent Linux vRealize Log Insight en tant qu'utilisateur racine ou non racine et vous pouvez définir les paramètres de configuration au cours de l'installation. Après l'installation, vous pouvez vérifier la version installée.

## Conditions préalables

- En savoir plus sur les valeurs d'installation par défaut et la façon de les modifier dans [Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux](#) .
- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- L'agent vRealize Log Insight Linux a besoin d'un accès au Syslog et aux services de mise en réseau pour fonctionner. Installez et exécutez l'agent Linux vRealize Log Insight sur les niveaux d'exécution 3 et 5. Si vous souhaitez que l'agent vRealize Log Insight Linux fonctionne sous d'autres niveaux d'exécution, configurez le système en conséquence.

## Procédure

1 Vous pouvez installer ou mettre à niveau un agent à partir de la console.

- Pour installer l'agent Linux vRealize Log Insight avec les paramètres de configuration par défaut, ouvrez une console et exécutez la commande suivante.

```
rpm -i VMware-Log-Insight-Agent-<version-and-build-number>.rpm
```

- Pour mettre à niveau l'agent sans modifier les paramètres de configuration actuels, ouvrez une console et exécutez la commande suivante.

```
rpm -Uhv VMware-Log-Insight-Agent-<version-and-build-number>.rpm
```

2 (Facultatif) Vous pouvez remplacer les valeurs de configuration par défaut pour l'installation ou les valeurs de configuration actuelles au cours d'une mise à jour. Pour cela, spécifiez les options dans le cadre de la commande d'installation ou de mise à niveau.

```
sudo <OPTION=value> rpm -i <version-and-build-number>.rpm
```

3 (Facultatif) Pour vérifier la version installée, exécutez la commande suivante.

```
rpm -qa | grep Log-Insight-Agent
```

## Exemple : Installation de l'agent Linux et exemples de mise à jour

- La commande suivante permet d'installer l'agent vRealize Log Insight pour une distribution Linux RPM. Elle installe l'agent sur un serveur distinct, attribue un numéro de port différent de celui par défaut, crée un utilisateur de l'agent vRealize Log Insight.

```
sudo SERVERHOST=myagentserver SERVERPORT=1234 LIAGENTUSER=liagent rpm -i VMware-Log-Insight-Agent-44.1234.rpm
```

- La commande suivante met à jour l'agent avec le fichier rpm donné. La configuration actuelle de l'agent n'est pas modifiée.

```
rpm -Uhv VMware-Log-Insight-Agent-44.1234.rpm
```

# Installer ou mettre à jour le module DEB de l'agent Linux vRealize Log Insight

Vous pouvez installer ou mettre à jour le module DEB (Debian) de l'agent Linux vRealize Log Insight à partir de la ligne de commande ou via la base de données debconf. Après l'installation, vous pouvez vérifier la version installée.

## Conditions préalables

- Pour en savoir plus sur les valeurs d'installation par défaut et la manière de les modifier, consultez la section [Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux](#).
- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Vérifiez que l'agent Linux vRealize Log Insight peut accéder aux services Syslog et aux services de mise en réseau afin de fonctionner. Par défaut, l'agent Linux vRealize Log Insight s'exécute aux niveaux d'exécution 2, 3, 4 et 5 et s'arrête aux niveaux d'exécution 0, 1 et 6.
- Pour plus d'informations et des exemples, reportez-vous à la section [Personnalisation de votre installation de l'agent pour Linux Debian](#).

## Procédure

- 1 Pour installer ou mettre à jour l'agent Linux vRealize Log Insight, ouvrez une console et exécutez la commande `dpkg -i package_name`.

*package\_name* se compose du préfixe **vmware-log-insight-agent-** et du numéro de build de votre version de téléchargement.

La commande au format suivant permet d'installer le module avec les valeurs par défaut.

```
dpkg -i vmware-log-insight-agent-VERSION-BUILD_NUMBER_all.deb
```

- 2 (Facultatif) Pour vérifier la version installée, exécutez la commande suivante :

```
dpkg -l | grep -i vmware-log-insight-agent
```

## Exemple

- Configurez le protocole de connexion à partir de la ligne de commande.

Pour remplacer le protocole de connexion par défaut, utilisez la variable `SERVERPROTO` comme indiqué dans l'exemple suivant :

```
sudo SERVERPROTO=syslog dpkg -i vmware-log-insight-agent-<version-and-build-number>_all.deb
```

## Personnalisation de votre installation de l'agent pour Linux Debian

Vous pouvez personnaliser votre installation en utilisant des options de commande pour remplacer les valeurs de configuration actuelles pour l'installation ou en configurant la base de données debconf.

### Personnalisation à partir de la ligne de commande

Pour configurer votre installation à partir de la ligne de commande, utilisez une commande au format suivant :

```
sudo <OPTION=value> dpkg -i vmware-log-insight-agent-<version-and-build-number>_all.deb
```

Pour obtenir la liste complète des options, reportez-vous à la section [Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux](#) .

Les exemples suivants montrent quelques configurations classiques faites à partir de la ligne de commande.

- Spécifiez un serveur vRealize Log Insight cible.
- Pour définir le serveur cible pendant l'installation, exécutez la commande `sudo` et remplacez `hostname` par l'adresse IP ou le nom d'hôte du serveur vRealize Log Insight comme indiqué dans l'exemple suivant :

```
sudo SERVERHOST=hostname dpkg -iv mware-log-insight-agent-<version-and-build-number>_all.deb
```

Si vous n'avez pas activé l'indicateur `--force-confold` au cours de l'installation, le système vous invite à conserver ou à remplacer le fichier de configuration `liagent.ini` à chaque fois que vous effectuez une mise à jour vers une version plus récente. Le message système suivant s'affiche :

```
Configuration file `/var/lib/loginsight-agent/liagent.ini'
==> Modified (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ? Your options are:
  Y or I : install the package maintainer's version
  N or O : keep your currently-installed version
  D      : show the differences between the versions
  Z      : start a shell to examine the situation
The default action is to keep your current version.
*** liagent.ini (Y/I/N/O/D/Z) [default=N] ?
```

Pour conserver la configuration existante, utilisez `[default=N]` . Les paramètres supplémentaires issus de la ligne de commande sont toujours appliqués.

- Configurer le protocole de connexion.

Pour remplacer le protocole de connexion par défaut, utilisez la variable `SERVERPROTO` comme indiqué dans l'exemple suivant :

```
sudo SERVERPROTO=syslog dpkg -i vmware-log-insight-agent-<version-and-build-number>_all.deb
```

- Configurez le port de connexion.

Pour remplacer le port de connexion par défaut, fournissez une valeur pour la variable `SERVERPORT` au programme d'installation comme indiqué dans l'exemple suivant :

```
sudo SERVERPORT=1234 dpkg -i vmware-log-insight-agent-<version-and-build-number>_all.deb
```

- Exécutez l'agent en tant qu'utilisateur non racine.

Pour exécuter l'agent Linux vRealize Log Insight en tant qu'utilisateur **non racine**, exécutez la commande `sudo`.

```
sudo LIAGENTUSER=liagent dpkg -i vmware-log-insight-agent-<version-build-number>_all.deb
```

Si l'utilisateur spécifié n'existe pas, l'agent Linux vRealize Log Insight crée le compte utilisateur pendant l'installation. Le compte créé n'est pas supprimé après la désinstallation. Si vous installez l'agent Linux avec le paramètre `LIAGENTUSER=non_root_user` et que vous tentez d'effectuer une mise à niveau à l'aide du paramètre `LIAGENTUSER=non_root_user2`, un conflit se produit. Des avertissements s'affichent, car l'utilisateur `non_root_user2` ne dispose pas des autorisations de l'utilisateur `non_root_user`.

## Options de personnalisation de module DEB pour la base de données debconf

Le module DEB de l'agent peut également être configuré via la base de données debconf. Le tableau suivant présente les options debconf prises en charge et celles correspondantes du programme d'installation DEB de l'agent vRealize Log Insight :

Options de ligne de commande	Options Debconf	Description
<code>SERVERHOST=hostname</code>	<code>vmware-log-insight-agent/serverhost</code>	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. La valeur par défaut est <b>loginsight</b> .
<code>SERVERPROTO={cfapi syslog}</code>	<code>vmware-log-insight-agent/serverproto</code>	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .

Options de ligne de commande	Options Debconf	Description
<code>SERVERPORT=<i>portnumber</i></code>	<code>vmware-log-insight-agent/serverport</code>	<p>Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut.</p> <ul style="list-style-type: none"> <li>■ cfapi avec SSL activé : 9543</li> <li>■ cfapi avec SSL désactivé : 9000</li> <li>■ syslog avec SSL activé : 6514</li> <li>■ syslog avec SSL désactivé : 514</li> </ul>
<code>LIAGENT_INITSYSTEM={<b>init</b> <b>systemd</b>}</code>	<code>log-insight-agent/init_system</code>	<p>Au moment de l'installation, l'agent détecte automatiquement le type de système de la machine sur laquelle vous installez l'agent. Vous pouvez remplacer ce comportement en spécifiant le type de valeur système avec cette option. Il existe deux types de systèmes init pris en charge : <code>init</code> et <code>systemd</code>.</p>
<code>LIAGENT_AUTOUPDATE={<b>yes</b> <b>no</b>}</code>	<code>vmware-log-insight-agent/auto_update</code>	<p>Active ou désactive la mise à jour automatique de l'agent. Vous devez également activer la mise à jour automatique à partir du serveur vRealize Log Insight pour activer la mise à jour automatique. La valeur par défaut est Yes.</p> <p>La mise à jour automatique n'est pas prise en charge pour les modules binaires Linux.</p>
<code>LI_AGENT_RUNSERVICES</code>	<code>vmware-log-insight-agent/init_system</code>	<p>Immédiatement après l'installation, les services <code>liagentd</code> (agent) et <code>liupdaterd</code> (mise à jour) sont démarrés par défaut. Vous pouvez empêcher leur démarrage en définissant le paramètre debconf <code>LIAGENT_RUNSERVICES</code> sur <code>no</code>. La valeur par défaut est <code>yes</code>. Les seules valeurs acceptées sont <b>yes</b> et <b>no</b> ; les valeurs <b>1</b> et <b>0</b> ne sont pas prises en charge.</p>

Options de ligne de commande	Options Debconf	Description
<code>LIAGENT_SSL</code>	<code>vmware-log-insight-agent/ssl</code>	C
<code>LIAGENTUSER=<i>user-account-name</i></code>	<code>vmware-log-insight-agent/liagentuser</code>	<p>Indique un compte sous lequel l'agent est exécuté. Si l'utilisateur n'existe pas, le programme d'installation crée un utilisateur normal. Si le compte d'utilisateur spécifié n'existe pas, l'agent Linux vRealize Log Insight crée le compte utilisateur pendant l'installation. Le compte créé n'est pas supprimé après la désinstallation.</p> <p>Par défaut, l'agent est configuré pour s'exécuter en tant qu'utilisateur racine.</p> <p>Si vous installez l'agent avec le paramètre <code>LIAGENTUSER=<i>non_root_user</i></code> et que vous tentez d'effectuer une mise à niveau avec <code>LIAGENTUSER=<i>non_root_user2</i></code>, un conflit survient. Des avertissements s'affichent, car l'utilisateur <i>non_root_user2</i> ne dispose pas des autorisations de l'utilisateur <i>non_root_user</i>.</p> <p>L'utilisateur créé n'est pas supprimé durant la désinstallation. Il peut être supprimé manuellement. Ce paramètre est destiné au service d'agent uniquement. Le service de mise à jour s'exécute toujours en tant qu'utilisateur racine.</p>

## Installer le module binaire Log Insight Linux Agent

L'installation du module binaire inclut la modification du fichier `.bin` en fichier exécutable et l'installation de l'agent.

La mise à niveau du package `.bin` n'est pas prise en charge officiellement. Si vous avez installé un Log Insight Linux Agent existant à l'aide du package `.bin`, effectuez une copie de sauvegarde du fichier `liagent.ini` situé dans le répertoire `/var/lib/loginsight-agent` afin de conserver la configuration locale. Après avoir réalisé la copie de sauvegarde, désinstallez manuellement le Log Insight Linux Agent. Reportez-vous à [Désinstaller manuellement le package bin de l'agent Linux Log Insight](#).

Si vous installez les agents Linux à l'aide du package `.bin`, le script `init.d` appelé `liagentd` est installé dans le cadre de l'installation du package, mais le package n'enregistre pas le script. Vous pouvez enregistrer le script manuellement.

Pour vérifier l'installation, exécutez la commande (`/sbin/`) `service liagentd status`.

### Conditions préalables

- Téléchargez et copiez le package Log Insight Linux Agent `.bin` vers la machine Linux cible.
- Vérifiez que le Log Insight Linux Agent a accès au Syslog et aux services de mise en réseau.

- Découvrez les valeurs de configuration par défaut et comment les modifier lors de l'installation. Reportez-vous à [Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux](#).

### Procédure

- 1 Ouvrez une console et exécutez la commande `chmod` pour modifier le fichier `.bin` en fichier exécutable.

Remplacez la *version du nom de fichier* par la version appropriée.

```
chmod +x filename-version.bin
```

- 2 À partir d'une invite de commande, exécutez la commande `./filename-version.bin` pour installer l'agent.

Remplacez la *version du nom de fichier* par la version appropriée.

```
./filename-version.bin
```

- 3 (Facultatif) Pour définir le serveur cible vRealize Log Insight au cours de l'installation, exécutez la commande `sudo SERVERHOST=hostname ./filename-version.bin`.

Remplacez *hostname* par l'adresse IP ou le nom d'hôte du serveur vRealize Log Insight.

```
sudo SERVERHOST=hostname ./filename-version.bin
```

- 4 (Facultatif) Pour remplacer le protocole de connexion par défaut, utilisez la variable `SERVERPROTO` comme indiqué dans l'exemple suivant :

```
sudo SERVERPROTO=syslog ./filename-version.htm
```

- 5 (Facultatif) Pour remplacer le port de connexion par défaut, fournissez une valeur pour la variable `SERVERPORT` au programme d'installation comme indiqué dans l'exemple suivant :

```
sudo SERVERPORT=1234 ./filename-version.htm
```

- 6 (Facultatif) Pour exécuter l'agent Log Insight Linux Agent en tant qu'utilisateur **non-racine**, exécutez la commande `sudo`.

```
sudo LIAGENTUSER=liagent ./filename-version.bin
```

Si l'utilisateur spécifié n'existe pas, Log Insight Linux Agent crée le compte utilisateur pendant l'installation. Le compte créé n'est pas supprimé après la désinstallation. Si vous installez le Log Insight Linux Agent avec le paramètre `LIAGENTUSER=non_root_user` et que vous tentez d'effectuer une mise à niveau à l'aide du paramètre `LIAGENTUSER=non_root_user2`, un conflit survient et des avertissements s'affichent, car l'utilisateur `non_root_user2` n'a pas les autorisations de l'utilisateur `non_root_user`.

## Options de ligne de commande pour l'installation de l'agent vRealize Log Insight sous Linux

Lorsque vous installez des agents vRealize Log Insight à partir de la ligne de commande, vous pouvez inclure des options pour configurer votre déploiement pendant l'installation. Ces options correspondent aux paramètres du fichier `liagent.ini`.

Les options suivantes peuvent être utilisées lors de l'installation afin de configurer les agents vRealize Log Insight qui s'exécutent sur des systèmes Linux.

Option	Description
<code>SERVERHOST=hostname</code>	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. La valeur par défaut est <code>loginsight</code> .
<code>SERVERPROTO={cfapi syslog}</code>	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .
<code>SERVERPORT=portnumber</code>	Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut. <ul style="list-style-type: none"> <li>■ <code>cfapi</code> avec SSL activé : 9543</li> <li>■ <code>cfapi</code> avec SSL désactivé : 9000</li> <li>■ <code>syslog</code> avec SSL activé : 6514</li> <li>■ <code>syslog</code> avec SSL désactivé : 514</li> </ul>
<code>LIAGENT_INITSYSTEM={init systemd}</code>	Au moment de l'installation, l'agent détecte automatiquement le type de système de la machine sur laquelle vous installez l'agent. Vous pouvez remplacer ce comportement en spécifiant le type de valeur système avec cette option. Il existe deux types de systèmes init pris en charge : <code>init</code> et <code>systemd</code> .
<code>LIAGENT_AUTOUPDATE={yes no}</code>	Active ou désactive la mise à jour automatique de l'agent. Vous devez également activer la mise à jour automatique à partir du serveur vRealize Log Insight pour activer la mise à jour automatique. La valeur par défaut est <code>Yes</code> . La mise à jour automatique n'est pas prise en charge pour les modules binaires Linux.

Option	Description
<code>LIAGENT_SSL={yes no}</code>	Activez une connexion sécurisée. Si SSL est activé, l'agent utilise le protocole TLS 1.2 pour communiquer avec le serveur. La valeur par défaut est Yes.
<code>LIAGENTUSER=<i>user-account-name</i></code>	<p>Indique un compte sous lequel l'agent est exécuté. Si l'utilisateur n'existe pas, le programme d'installation crée un utilisateur normal. Si le compte d'utilisateur spécifié n'existe pas, l'agent Linux vRealize Log Insight crée le compte utilisateur pendant l'installation. Le compte créé n'est pas supprimé après la désinstallation.</p> <p>Par défaut, l'agent est configuré pour s'exécuter en tant qu'utilisateur racine.</p> <p>Si vous installez avec le paramètre <code>LIAGENTUSER=<i>non_root_user</i></code> et que vous tentez d'effectuer une mise à niveau avec <code>LIAGENTUSER=<i>non_root_user2</i></code>, un conflit survient. Des avertissements s'affichent, car l'utilisateur <code><i>non_root_user2</i></code> ne dispose pas des autorisations de l'utilisateur <code><i>non_root_user</i></code>.</p> <p>L'utilisateur créé n'est pas supprimé durant la désinstallation. Il peut être supprimé manuellement. Ce paramètre est destiné au service d'agent uniquement. Le service de mise à jour s'exécute toujours en tant qu'utilisateur racine.</p>

## Mise à jour automatique des agents vRealize Log Insight

La fonctionnalité de mise à jour automatique des agents vRealize Log Insight permet aux agents actifs d'effectuer une vérification, un téléchargement et une mise à jour automatique en fonction des modules installés sur l'agent, à partir du serveur vRealize Log Insight.

Vous pouvez activer la mise à jour automatique à partir du serveur pour tous les agents ou à partir des clients pour certaines instances d'agent spécifiques. Elle s'applique aux agents qui sont à l'état Actif et dont la version est 4.3 ou ultérieure.

La mise à jour automatique n'est pas prise en charge pour les modules binaires Linux.

### Activer ou désactiver la mise à jour automatique pour des agents spécifiques

Vous pouvez activer ou désactiver la mise à jour automatique pour des agents spécifiques en modifiant le fichier de configuration côté client de l'agent en question.

Par défaut, la mise à jour automatique d'un agent est activée côté client.

#### Conditions préalables

La version des agents doit être 4.3 ou une version ultérieure

## Procédure

- 1 Ouvrez le fichier `liagent.ini` local dans un éditeur.
- 2 Repérez la section `[update]`.

Elle ressemble à l'exemple ci-dessous.

```
[update]
; Do not change this parameter
package_type=msi
; Enable automatic update of the agent. If enabled:
; the agent will silently check for updates from the server and
; if available will automatically download and apply the update.
; auto_update=yes
```

- 3 Pour désactiver la mise à jour automatique, annulez le commentaire de `auto_update=yes` et remplacez-le par `auto_update=no`.

---

**Note** La mise à jour automatique des agents est activée par défaut. Par conséquent, la valeur par défaut de `auto_update` est « oui », même si elle est commentée.

---

- 4 Enregistrez et fermez le fichier `liagent.ini`.

# Configuration des agents vRealize Log Insight

# 4

Après avoir déployé un agent, vous pouvez notamment le configurer pour qu'il envoie des événements au serveur vRealize Log Insight de votre choix et vous pouvez spécifier des protocoles de communication et définir d'autres paramètres.

Suivez ces instructions pour configurer les agents selon vos besoins.

- [Configurer Log Insight Windows Agent](#)

Vous pouvez configurer Log Insight Windows Agent après l'avoir installé. Modifiez le fichier `liagent.ini` de manière à configurer Log Insight Windows Agent pour qu'il envoie les événements à un serveur vRealize Log Insight, définisse le protocole et le port de communication, ajoute des canaux d'événements Windows, et configure la collecte de journaux de fichiers plats. Le fichier se trouve dans le répertoire `%ProgramData%\VMware\Log Insight Agent`.

- [Configurer Log Insight Linux Agent](#)

Vous pouvez configurer Log Insight Linux Agent après l'avoir installé.

- [Filtrage des événements provenant des agents vRealize Log Insight](#)

Vous pouvez fournir les informations qu'un agent envoie à une destination avec l'option de filtre dans la section `[server|<dest_id>]` du fichier local `liagent.ini`.

- [Transfert d'informations à partir d'un agent vRealize Log Insight](#)

Vous pouvez transférer des événements collectés par un agent vers trois destinations au maximum. Une destination peut inclure des serveurs ou un redirecteur vRealize Log Insight, ou bien des solutions tierces de gestion des journaux.

- [Configuration centralisée des agents vRealize Log Insight](#)

Vous pouvez configurer plusieurs agents vRealize Log Insight.

- [Utilisation de valeurs communes pour la configuration des agents](#)

Vous pouvez remplacer les valeurs par défaut du fichier de configuration de l'agent par des valeurs de paramètre communes qui s'appliquent à la section de configuration de chaque agent pour les agents Windows ou Linux.

## ■ Analyse des journaux

Les analyseurs de journaux provenant d'agents extraient les données structurées de journaux bruts avant de les livrer au serveur vRealize Log Insight. Grâce à ces analyseurs, vRealize Log Insight peut analyser des journaux, en extraire des informations et afficher ces résultats sur le serveur. Vous pouvez configurer les analyseurs de journaux pour les agents vRealize Log Insight.

## Configurer Log Insight Windows Agent

Vous pouvez configurer Log Insight Windows Agent après l'avoir installé. Modifiez le fichier `liagent.ini` de manière à configurer Log Insight Windows Agent pour qu'il envoie les événements à un serveur vRealize Log Insight, définisse le protocole et le port de communication, ajoute des canaux d'événements Windows, et configure la collecte de journaux de fichiers plats. Le fichier se trouve dans le répertoire `% ProgramData\VMware\Log Insight Agent`.

## Configuration par défaut de Log Insight Windows Agent

Après l'installation, le fichier `liagent.ini` contient des paramètres par défaut préconfigurés pour Log Insight Windows Agent.

## Configuration par défaut du fichier `liagent.ini` de Log Insight Windows Agent

Si vous utilisez des noms et des valeurs non-ASCII, enregistrez la configuration en UTF-8.

Si vous utilisez une configuration centrale, la configuration finale est ce fichier joint avec des paramètres du serveur pour former le fichier `liagent-effective.ini`.

Vous trouverez éventuellement plus efficace de configurer les paramètres dans la page des agents du serveur.

```
; Client-side configuration of VMware Log Insight Agent.
; See liagent-effective.ini for the actual configuration used by VMware Log Insight Agent.

[server]
; Log Insight server hostname or ip address
; If omitted the default value is LOGINSIGHT
;hostname=LOGINSIGHT

;Enables or disables centralized configuration from the vRealize Log Insight server.
;When enabled, agent configuration changes made to the liagent.ini file on the server
;are joined with the settings in this file. to this agent. Accepted values are yes or no and
0 or 1.
;The default is yes.
;
;central_config=yes
;

; Set protocol to use:
; cfapi - Log Insight REST API
; syslog - Syslog protocol
; If omitted the default value is cfapi
```

```

;
;proto=cfapi

; Log Insight server port to connect to. If omitted the default value is:
; for syslog: 514
; for cfapi without ssl: 9000
; for cfapi with ssl: 9543
;port=9000

;ssl - enable/disable SSL. Applies to cfapi protocol only.
; Possible values are yes or no. If omitted the default value is no.
;ssl=no

; Time in minutes to force reconnection to the server
; If omitted the default value is 30
;reconnect=30

[storage]
;max_disk_buffer - max disk usage limit (data + logs) in MB:
; 100 - 2000 MB, default 200
;max_disk_buffer=200

[logging]
;debug_level - the level of debug messages to enable:
; 0 - no debug messages
; 1 - trace essential debug messages
; 2 - verbose debug messages (will have negative impact on performance)
;debug_level=0
;
;The interval in minutes to print statistics
;stats_period=15

[update]
; Do not change this parameter
package_type=msi

; Enable automatic update of the agent. If enabled:
; the agent will silently check for updates from the server and
; if available will automatically download and apply the update.
;auto_update=yes

[winlog|Application]
channel=Application
raw_syslog=no

[winlog|Security]
channel=Security

[winlog|System]
channel=System

```

Paramètre	Valeur par défaut	Description
hostname	LOGINSIGHT	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. La valeur par défaut est <b>loginsight</b> .
central_config	yes	Activez ou désactivez la configuration centralisée pour cet agent. Lorsque la configuration centralisée est désactivée, l'agent ignore la configuration fournie par le serveur vRealize Log Insight. Les valeurs acceptées sont <code>yes</code> , <code>no</code> , <code>1</code> ou <code>0</code> . La valeur par défaut est <code>yes</code> .
proto	cfapi	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .
port	9543, 9000, 6514 et 514	Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut. <ul style="list-style-type: none"> <li>■ cfapi avec SSL activé : 9543</li> <li>■ cfapi avec SSL désactivé : 9000</li> <li>■ syslog avec SSL activé : 6514</li> <li>■ syslog avec SSL désactivé : 514</li> </ul>
ssl	yes	Active ou désactive SSL. La valeur par défaut est oui. Lorsque <code>ssl</code> est défini sur oui, si vous ne définissez pas de valeur pour le port, le port 9543 est celui choisi automatiquement.
max_disk_buffer	200	Espace disque maximal utilisé par Log Insight Windows Agent (en Mo) pour mettre les événements et ses propres journaux en mémoire tampon. Lorsque la valeur maximale spécifiée pour <code>max_disk_buffer</code> est atteinte, l'agent commence à annuler les nouveaux événements entrants.

Paramètre	Valeur par défaut	Description
debug_level	0	Définit le niveau de détail des journaux. Reportez-vous à <a href="#">Définir le niveau de détail des journaux dans le Log Insight Agents</a> .
channel	Application, sécurité, système	Les canaux Application, Sécurité et Système du journal des événements Windows sont mis en commentaire par défaut ; Log Insight Windows Agent ne collecte pas les journaux de ces canaux.  Reportez-vous à <a href="#">Collecter des événements provenant de canaux d'événements Windows</a> .
raw_syslog	no	Pour les agents utilisant le protocole Syslog, autorise l'agent à collecter et à envoyer des événements Syslog bruts. La valeur par défaut est No, ce qui signifie que les événements collectés sont transformés par des attributs Syslog spécifiés par l'utilisateur. Activez cette option pour collecter des événements sans transformations syslog.  Les valeurs acceptées sont Oui ou 1, et Non ou 0.

## Collecter des événements provenant de canaux d'événements Windows

Vous pouvez ajouter un canal d'événements Windows à la configuration de Log Insight Windows Agent. Log Insight Windows Agent collecte les événements et les envoie au serveur vRealize Log Insight.

Les champs de noms sont limités. Les noms suivants sont réservés et ne peuvent pas être utilisés comme noms de champs.

- event\_type
- hostname
- source
- text

### Conditions préalables

Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

## Procédure

- 1 Accédez au répertoire des données du programme de l'agent vRealize Log Insight Windows.

```
%ProgramData%\VMware\Log Insight Agent
```

- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.

- 3 Ajoutez les paramètres suivants et définissez les valeurs pour votre environnement.

Paramètre	Description
<code>[winlog section_name]</code>	Nom unique de la section de configuration.
<code>channel</code>	Nom complet du canal d'événements, comme indiqué dans l'Observateur d'événements, application Windows intégrée. Pour copier le nom de canal correct, cliquez avec le bouton droit sur un canal dans l'Observateur d'événements, sélectionnez <b>Propriétés</b> et copiez le contenu du champ <b>Nom complet</b> .
<code>enabled</code>	Paramètre facultatif permettant d'activer ou de désactiver la section de configuration. Les valeurs possibles sont oui ou non (non sensibles à la casse). La valeur par défaut est oui.
<code>tags</code>	Paramètre facultatif permettant d'ajouter des balises personnalisées aux champs d'événements collectés. Définissez des balises avec la notation JSON. Les noms de balises peuvent contenir des lettres, des chiffres et des traits de soulignement. Un nom de balise ne peut commencer que par une lettre ou un trait de soulignement, et ne doit pas dépasser 64 caractères. Les noms de balise ne sont pas sensibles à la casse. Par exemple, si vous utilisez <code>tags={"nom_balise1" : "valeur balise 1", "Nom_Balise1" : "valeur balise 2" }</code> , <code>Nom_Balise1</code> est ignoré, car il est en double. Vous ne pouvez pas utiliser <code>event_type</code> et <code>timestamp</code> comme noms de balises. Tous les doublons inclus dans la même déclaration sont ignorés.  Si la destination est un serveur Syslog, les balises peuvent remplacer le champ APP-NAME. Par exemple, <code>tags={"appname":"VROPS"}</code> .
<code>whitelist, blacklist</code>	Paramètres facultatifs permettant d'inclure ou d'exclure explicitement les événements de journaux.  <b>Note</b> L'option <code>blacklist</code> fonctionne uniquement pour les champs ; elle ne peut pas être utilisée pour bloquer le texte.
<code>exclude_fields</code>	(Facultatif) Paramètre permettant d'exclure des champs donnés d'une collecte. Vous pouvez fournir plusieurs valeurs sous la forme d'une liste séparée par des points-virgules. Par exemple, <code>exclude_fields=EventId; ProviderName</code>

```
[winlog|section_name]
channel=event_channel_name
enabled=yes_or_no
tags={"tag_name1" : "Tag value 1", "tag_name2" : "tag value 2" }
```

- 4 Enregistrez et fermez le fichier `liagent.ini`.

## Exemple : Configurations

Reportez-vous aux exemples de configuration de `[winlog|` fournis ci-dessous.

```
[winlog|Events_Firewall ]
channel=Microsoft-Windows-Windows Firewall With Advanced Security/Firewall
enabled=no
```

```
[winlog|custom]
channel=Custom
tags={"ChannelDescription": "Events testing channel"}
```

## Configurer le filtrage des canaux d'événements Windows

Vous pouvez configurer le filtrage des canaux d'événements Windows afin d'inclure ou d'exclure des événements de journaux de façon explicite.

Vous pouvez utiliser les paramètres `whitelist` et `blacklist` pour évaluer une expression de filtre. L'expression de filtre est une expression booléenne constituée de champs d'événements et d'opérateurs.

---

**Note** L'option `blacklist` fonctionne uniquement pour les champs ; elle ne peut pas être utilisée pour bloquer le texte.

---

- Le paramètre `whitelist` ne collecte que les événements de journaux pour lesquels l'expression de filtre indique une valeur différente de zéro. Si vous omettez ce paramètre, la valeur est un 1 implicite.
- Le paramètre `blacklist` exclut les événements du journal pour lesquels l'expression du filtre indique une valeur non nulle. La valeur par défaut est 0.

Pour connaître la liste complète des champs d'événements et des opérateurs Windows, reportez-vous à [Champs d'événements et opérateurs](#).

### Conditions préalables

Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

### Procédure

- 1 Accédez au répertoire des données du programme de l'agent vRealize Log Insight Windows.  
`%ProgramData%\VMware\Log Insight Agent`
- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.

### 3 Ajoutez un paramètre `whitelist` ou `blacklist` à la section `[winlog|]`.

Par exemple

```
[winlog|unique_section_name]
channel = event_channel_name
blacklist = filter_expression
```

### 4 Créez une expression de filtre à partir de champs d'événements et d'opérateurs Windows.

Par exemple

```
whitelist = level > WINLOG_LEVEL_SUCCESS and level < WINLOG_LEVEL_INFO
```

### 5 Enregistrez et fermez le fichier `liagent.ini`.

#### Exemple : Configurations de filtres

Vous pouvez configurer l'agent pour qu'il collecte uniquement les événements d'erreur, par exemple

```
[winlog|Security-Error]
channel = Security
whitelist = Level == WINLOG_LEVEL_CRITICAL or Level == WINLOG_LEVEL_ERROR
```

Vous pouvez configurer l'agent pour qu'il collecte uniquement les événements Réseau VMware du canal Application, par exemple

```
[winlog|VMwareNetwork]
channel = Application
whitelist = ProviderName == "VMnetAdapter" or ProviderName == "VMnetBridge" or ProviderName == "VMnetDHCP"
```

Vous pouvez configurer l'agent pour qu'il collecte tous les événements du canal Sécurité, sauf des événements particuliers, par exemple

```
[winlog|Security-Verbose]
channel = Security
blacklist = EventID == 4688 or EventID == 5447
```

## Champs d'événements et opérateurs

Utilisez les champs d'événements et les opérateurs de Windows pour générer des expressions de filtre.

### Opérateurs d'expression de filtre

Opérateur	Description
<code>==, !=</code>	égal à et différent de. À utiliser avec des champs numériques et des champs de chaînes.
<code>&gt;, &gt;, &lt;, &lt;=</code>	supérieur ou égal à, supérieur à, inférieur à, inférieur ou égal à. À utiliser uniquement avec des champs numériques.

Opérateur	Description
&,  , ^, ~	Opérations AND, OR, XOR au niveau du bit et opérateurs de complément. À utiliser uniquement avec des champs numériques.
and, or	AND et OR logiques. À utiliser pour générer des expressions complexes en combinant des expressions simples.
pas	Opérateur logique NOT unaire. À utiliser pour inverser la valeur d'une expression.
()	Utilisez des parenthèses dans une expression logique pour modifier l'ordre d'évaluation.

## Champs d'événements Windows

Vous pouvez utiliser les champs d'événements Windows suivants dans une expression de filtre.

Nom du champ	Type de champ
Adresse Internet	string
Texte	string
ProviderName	string
EventSourceName	string
EventID	numeric
EventRecordID	numeric
Canal	string
UserID	string
Niveau	numeric Vous pouvez utiliser les constantes prédéfinies suivantes. <ul style="list-style-type: none"> <li>■ WINLOG_LEVEL_SUCCESS = 0</li> <li>■ WINLOG_LEVEL_CRITICAL = 1</li> <li>■ WINLOG_LEVEL_ERROR = 2</li> <li>■ WINLOG_LEVEL_WARNING = 3</li> <li>■ WINLOG_LEVEL_INFO = 4</li> <li>■ WINLOG_LEVEL_VERBOSE = 5</li> </ul>
Tâche	numeric
OpCode	numeric
Mots clés	numeric Vous pouvez utiliser les masques de bits prédéfinis suivants. <ul style="list-style-type: none"> <li>■ WINLOG_KEYWORD_RESPONSETIME = 0x0001000000000000;</li> <li>■ WINLOG_KEYWORD_WDICONTEXT = 0x0002000000000000;</li> <li>■ WINLOG_KEYWORD_WDIDIAGNOSTIC = 0x0004000000000000;</li> <li>■ WINLOG_KEYWORD_SQM = 0x0008000000000000;</li> <li>■ WINLOG_KEYWORD_AUDITFAILURE = 0x0010000000000000;</li> <li>■ WINLOG_KEYWORD_AUDITSUCCESS = 0x0020000000000000;</li> <li>■ WINLOG_KEYWORD_CORRELATIONHINT = 0x0040000000000000;</li> <li>■ WINLOG_KEYWORD_CLASSIC = 0x0080000000000000;</li> </ul>

## Exemples

Collecte de tous les événements critiques, d'erreur et d'avertissement

```
[winlog|app]
channel = Application
whitelist = level > WINLOG_LEVEL_SUCCESS and level < WINLOG_LEVEL_INFO
```

Collecte uniquement des événements d'erreur d'audit à partir du canal de sécurité

```
[winlog|security]
channel = Security
whitelist = Keywords & WINLOG_KEYWORD_AUDITFAILURE
```

## Collecter les événements d'un fichier journal

Vous pouvez configurer l'agent vRealize Log Insight Windows afin de collecter des événements à partir d'un ou de plusieurs fichiers journaux.

Les champs de noms sont limités. Les noms suivants sont réservés et ne peuvent pas être utilisés comme noms de champs.

- event\_type
- hostname
- source
- text

Vous pouvez avoir jusqu'à trois destinations pour les informations de l'agent et filtrer les informations avant leur envoi. Reportez-vous à [Transfert d'informations à partir d'un agent vRealize Log Insight](#).

---

### Note

- La surveillance d'un grand nombre de fichiers, par exemple un millier ou plus, entraîne une utilisation plus élevée des ressources par l'agent et affecte les performances globales de la machine hôte. Pour éviter cela, configurez l'agent pour qu'il surveille uniquement les fichiers nécessaires à l'aide de modèles et de caractères génériques ou qu'il archive les anciens fichiers journaux. S'il n'est pas nécessaire de surveiller un grand nombre de fichiers, augmentez les paramètres de l'hôte, tels que le CPU et la RAM.
  - L'agent peut effectuer la collecte depuis des répertoires chiffrés, mais uniquement en cas d'exécution par l'utilisateur qui a chiffré le répertoire.
  - L'agent ne prend en charge que les structures de répertoires statiques. Si les répertoires ont été renommés ou ajoutés, vous devez redémarrer l'agent pour commencer à surveiller ces répertoires, à condition que la configuration couvre ces répertoires.
-

## Conditions préalables

Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

## Procédure

- 1 Accédez au répertoire des données du programme de l'agent vRealize Log Insight Windows.  
`%ProgramData%\VMware\Log Insight Agent`
- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.
- 3 Localisez la section `[server|<dest_id>]` du fichier. Ajoutez des paramètres de configuration et définissez les valeurs de votre environnement.

```
[filelog|section_name]
directory=path_to_log_directory
include=glob_pattern
...
```

Paramètre	Description
<code>[filelog section_name]</code>	Nom unique de la section de configuration.
<code>directory=full-path-to-log-file</code>	<p>Chemin d'accès complet au répertoire du fichier journal. Les modèles comportant des caractères génériques sont pris en charge. Exemples de configurations :</p> <ul style="list-style-type: none"> <li>■ Pour effectuer une collecte à partir de tous les sous-répertoires du répertoire <code>D:\Logs\new_test_logs</code>, utilisez <code>directory=D:\Logs\new_test_logs\*</code></li> <li>■ Si vos sous-répertoires disposent de leurs propres sous-répertoires, utilisez la configuration suivante pour surveiller tous les sous-répertoires <code>directory=D:\Logs\new_test_logs\*\*</code></li> </ul> <p><b>Note</b> Pour limiter le nombre de fichiers et de répertoires et éviter une consommation de ressources élevée, vous ne pouvez pas définir un répertoire glob pour les répertoires de premier ou de deuxième niveau, tels que <code>directory=c:/tmp/*</code> ou <code>directory=c:\Logs\*</code>. Le chemin au répertoire doit être d'au moins deux niveaux.</p> <p>Vous pouvez définir un chemin vers un répertoire non existant, et l'agent collectera les fichiers journaux dans ce répertoire une fois le répertoire et les fichiers créés.</p> <p>Vous pouvez définir le même répertoire pour une ou plusieurs sections de configuration différentes, ceci afin de collecter plusieurs fois les journaux depuis le même fichier. Ce processus rend possible l'application de différents filtres et différentes balises à la même source d'événements.</p> <p><b>Note</b> Si vous utilisez des configurations identiques pour ces sections, les événements en double apparaissent du côté serveur.</p>

Paramètre	Description
<code>include= file_name; ...</code>	<p>(Facultatif) Nom de fichier ou masque de fichier (modèle glob) à partir duquel collecter des données. Vous pouvez fournir des valeurs sous la forme d'une liste séparée par des points-virgules. La valeur par défaut est <code>*</code> qui prévoit l'inclusion de tous les fichiers. Le paramètre respecte la casse.</p> <p>Un masque de fichier (modèle glob) peut être utilisé pour grouper les fichiers qui suivent la même convention de dénomination, ainsi que dans un nom de fichier unique. Par exemple, les noms de fichier qui contiennent des espaces, tels que <code>vRealize Ops Analytics.log</code> et <code>vRealize Ops Collector.log</code>, peuvent être spécifiés avec <code>vRealize vRealize?Ops?Analytics*.log</code> ou <code>vRealize*.log</code>. En utilisant des masques de fichier, vous pouvez spécifier des noms de fichier qui sont acceptables pour la configuration de l'agent sous des hôtes Linux et Windows.</p> <p>Les fichiers <code>.zip</code> et <code>.gz</code> sont exclus par défaut de la collecte.</p> <p><b>Important</b> Si vous collectez un fichier journal archivé par rotation, utilisez les paramètres <code>include</code> et <code>exclude</code> pour spécifier un modèle générique correspondant au fichier principal et au fichier archivé par rotation. Si le modèle générique correspond uniquement au fichier journal principal, les agents vRealize Log Insight peuvent ne pas prendre en compte des événements lors de l'archivage par rotation. Les agents vRealize Log Insight déterminent automatiquement l'ordre correct des fichiers archivés par rotation et envoient les événements au serveur vRealize Log Insight dans l'ordre approprié. Par exemple, si votre fichier journal principal se nomme <code>myapp.log</code> et si les journaux archivés par rotation sont <code>myapp.log.1</code>, <code>myapp.log.2</code> etc., vous pouvez utiliser le modèle <code>include</code> suivant :</p> <pre>include= myapp.log;myapp.log.*</pre>
<code>exclude=regular_expression</code>	<p>(Facultatif) Nom ou masque de fichier (modèle générique) à exclure de la collecte. Vous pouvez fournir des valeurs sous la forme d'une liste séparée par des points-virgules. La liste est vide par défaut, ce qui signifie qu'aucun fichier n'est exclu.</p>
<code>event_marker=regular_expression</code>	<p>(Facultatif) Expression régulière qui désigne le début d'un événement dans le fichier journal. En cas d'omission, la valeur par défaut est nouvelle ligne. Les expressions que vous tapez doivent utiliser la syntaxe des expressions régulières Perl.</p> <p><b>Note</b> Les symboles, par exemple les guillemets (" "), ne sont pas traités comme des wrappers pour les expressions régulières. Ils sont traités comme partie intégrante du modèle.</p> <p>Étant donné que l'agent vRealize Log Insight est optimisé pour la collecte en temps réel, les messages journaux partiels écrits avec un délai interne peuvent être fractionnés en plusieurs événements. Si l'ajout de fichiers journaux s'arrête pendant plus de 200 ms sans nouvel <code>event_marker</code>, l'événement partiel est traité comme terminé, analysé et livré. Cette logique de minutage est non configurable et prioritaire par rapport au paramètre <code>event_marker</code>. Les appenders de fichiers journaux doivent purger les événements complets.</p>
<code>enabled=yes no</code>	<p>(Facultatif) Paramètre permettant d'activer ou de désactiver la section de configuration. Les valeurs possibles sont <code>yes</code> ou <code>no</code>. La valeur par défaut est <code>yes</code>.</p>

Paramètre	Description
<b>charset=char-encoding-type</b>	(Facultatif) Codage de caractères des fichiers journaux que surveille l'agent. Les valeurs possibles sont : <ul style="list-style-type: none"> <li>■ UTF-8</li> <li>■ UTF-16LE</li> <li>■ UTF-16BE</li> </ul> La valeur par défaut est UTF-8.
<b>tags={"tag-name" : "tag-value", ...}</b>	(Facultatif) Paramètre permettant d'ajouter des balises personnalisées aux champs d'événements collectés. Définissez des balises avec la notation JSON. Les noms de balises peuvent contenir des lettres, des chiffres et des traits de soulignement. Un nom de balise ne peut commencer que par une lettre ou un trait de soulignement, et ne doit pas dépasser 64 caractères. Les noms de balise ne sont pas sensibles à la casse. Par exemple, si vous utilisez tags={"nom_balise1" : "valeur balise 1", "Nom_Balise1" : "valeur balise 2" }, Nom_Balise1 est ignoré, car il est en double. Vous ne pouvez pas utiliser event_type et timestamp comme noms de balises. Tous les doublons inclus dans la même déclaration sont ignorés.  Si la destination est un serveur Syslog, les balises peuvent remplacer le champ APP-NAME. Par exemple, tags={"appname": "VROPS"}.
<b>exclude_fields</b>	(Facultatif) Paramètre permettant d'exclure des champs donnés d'une collecte. Vous pouvez fournir plusieurs valeurs sous forme de liste séparée par des points-virgules ou des virgules. Par exemple, <ul style="list-style-type: none"> <li>■ exclude_fields=hostname; filepath</li> <li>■ exclude_fields=type; size</li> <li>■ exclude_fields=type, size</li> </ul>
<b>raw_syslog=Yes No</b>	Pour les agents utilisant le protocole Syslog, cette option autorise l'agent à collecter et à envoyer des événements Syslog bruts. La valeur par défaut est Non, ce qui signifie que les événements collectés sont transformés par des attributs Syslog spécifiés par l'utilisateur. Activez cette option pour collecter des événements sans transformations syslog.

## Exemple : Configurations

```
[filelog|vCenterMain]
directory=C:\ProgramData\VMware\VMware VirtualCenter\Logs
include=vpxd-*.log
exclude=vpxd-alert-*.log;vpxd-profiler-*.log
event_marker=^{\d{4}}-\d{2}-\d{2} [A-Z] \d{2}:\d{2}:\d{2}\.\d{3}
```

```
[filelog|ApacheAccessLogs]
enabled=yes
directory=C:\Program Files (x86)\Apache Software Foundation\Apache2.2\logs
include=*.log
exclude=*_old.log
tags={"Provider" : "Apache"}
```

```
[filelog|MSSQL]
directory=C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\Log
```

```
charset=UTF-16LE
event_marker=[^\s]
```

## Configurer le filtrage des canaux de fichiers journaux Windows

Vous pouvez configurer le filtrage des fichiers journaux Windows afin d'inclure ou d'exclure des événements de journaux de façon explicite.

Vous pouvez utiliser les paramètres `whitelist` et `blacklist` pour évaluer une expression de filtre. L'expression de filtre est une expression booléenne constituée de champs d'événements et d'opérateurs.

---

**Note** L'option `blacklist` fonctionne uniquement pour les champs ; elle ne peut pas être utilisée pour bloquer le texte.

---

- Le paramètre `whitelist` ne collecte que les événements de journaux pour lesquels l'expression de filtre indique une valeur différente de zéro. Si vous omettez ce paramètre, la valeur est un 1 implicite.
- Le paramètre `blacklist` exclut les événements du journal pour lesquels l'expression du filtre indique une valeur non nulle. La valeur par défaut est 0.

Pour connaître la liste complète des champs d'événements et des opérateurs Windows, reportez-vous à [Champs d'événements et opérateurs](#).

### Conditions préalables

Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

### Procédure

- 1 Accédez au répertoire des données du programme de l'agent vRealize Log Insight Windows.  
`%ProgramData%\VMware\Log Insight Agent`
- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.
- 3 Ajoutez un paramètre `whitelist` ou `blacklist` à la section `[filelog|]`.

Par exemple :

```
[filelog|apache]
directory = path_to_log_directory
include = glob_pattern
blacklist = filter_expression
```

- 4 Créez une expression de filtre à partir de champs d'événements et d'opérateurs Windows.

Par exemple

```
whitelist = myServer
```

5 Enregistrez et fermez le fichier `liagent.ini`.

### Exemple : Configurations de filtres

Vous pouvez configurer l'agent pour qu'il ne collecte que les journaux Apache où le nom de serveur est

```
[filelog|apache]
directory=C:\Program Files\Apache Software Foundation\Apache2.4\logs
include=error.log
parser=clf
whitelist = server_name == "sample.com"
blacklist = remote_host == "127.0.0.1"
```

## Transférer des événements vers Log Insight Windows Agent

Vous pouvez transférer des événements de machines Windows vers une machine sur laquelle Log Insight Windows Agent est en cours d'exécution.

Vous pouvez utiliser le transfert d'événements de Windows pour transférer des événements de plusieurs machines Windows vers une machine sur laquelle Log Insight Windows Agent est installé. Vous pouvez ensuite configurer Log Insight Windows Agent afin qu'il collecte tous les événements transférés et les envoie à un serveur vRealize Log Insight.

Familiarisez-vous avec le transfert d'événements de Windows. Reportez-vous à <http://technet.microsoft.com/en-us/library/cc748890.aspx> et à [http://msdn.microsoft.com/en-us/library/windows/desktop/bb870973\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/bb870973(v=vs.85).aspx).

### Conditions préalables

Reportez-vous à [Collecter des événements provenant de canaux d'événements Windows](#) .

### Procédure

- 1 Ajoutez une nouvelle section à la configuration de Log Insight Windows Agent pour collecter des événements du canal d'événements Windows qui reçoit les événements transférés.

Le nom de canal par défaut est `ForwardedEvents`.

- 2 Configurez le transfert d'événements de Windows.

### Étape suivante

Accédez à l'interface utilisateur Web de vRealize Log Insight et vérifiez que les événements transférés arrivent.

## Configurer Log Insight Linux Agent

Vous pouvez configurer Log Insight Linux Agent après l'avoir installé.

Vos pouvez utiliser la [configuration d'agent centralisée](#) pour configurer l'agent afin d'envoyer les événements à un serveur vRealize Log Insight, de spécifier le port et le protocole de communication et de configurer la collecte des fichiers journaux plats. Pour obtenir l'emplacement du fichier `liagent.ini`, reportez-vous à [Définir le niveau de détail des journaux dans le Log Insight Agents](#).

## Configuration par défaut de l'agent Linux vRealize Log Insight

Après l'installation, le fichier `liagent.ini` contient des paramètres par défaut préconfigurés pour Log Insight Windows Agent.

### Configuration par défaut du fichier `liagent.ini` de l'agent Linux vRealize Log Insight

Si vous utilisez des noms et des valeurs non-ASCII, enregistrez la configuration en UTF-8.

Si vous utilisez une configuration centrale, la configuration finale est ce fichier joint avec des paramètres du serveur pour former le fichier `liagent-effective.ini`.

Vous trouverez éventuellement plus efficace de configurer les paramètres dans la page des agents du serveur.

```
[server]
; Hostname or IP address of your Log Insight server / cluster load balancer. Default:
;hostname=LOGINSIGHT

;Enables or disables centralized configuration from the vRealize Log Insight server.
;When enabled, agent configuration changes made to the liagent.ini file on the server
;are joined with the settings in this file. to this agent. Accepted values are yes or no and
0 or 1.
;The default is yes.
;
;central_config=yes
;
;
; Protocol can be cfapi (Log Insight REST API), syslog. Default:
;proto=cfapi

; Log Insight server port to connect to. Default ports for protocols (all TCP):
; syslog: 514; syslog with ssl: 6514; cfapi: 9000; cfapi with ssl: 9543. Default:
;port=9543

; SSL usage. Default:
;ssl=yes
; Example of configuration with trusted CA:
;ssl=yes
;ssl_ca_path=/etc/pki/tls/certs/ca.pem

; Time in minutes to force reconnection to the server.
; This option mitigates imbalances caused by long-lived TCP connections. Default:
;reconnect=30
```

```
[logging]
; Logging verbosity: 0 (no debug messages), 1 (essentials), 2 (verbose with more impact on
performance).
; This option should always be 0 under normal operating conditions. Default:
;debug_level=0

[storage]
; Max local storage usage limit (data + logs) in MBs. Valid range: 100-2000 MB.
;max_disk_buffer=200

; Uncomment the appropriate section to collect system logs
; The recommended way is to enable the Linux content pack from LI server
;[filelog|syslog]
;directory=/var/log
;include=messages;messages.?.syslog;syslog.?
```

Paramètre	Valeur par défaut	Description
hostname	LOGINSIGHT	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. La valeur par défaut est <b>loginsight</b> .
central_config	yes	Activez ou désactivez la configuration centralisée pour cet agent. Lorsque la configuration centralisée est désactivée, l'agent ignore la configuration fournie par le serveur vRealize Log Insight. Les valeurs acceptées sont <code>yes</code> , <code>no</code> , <code>1</code> ou <code>0</code> . La valeur par défaut est <code>yes</code> .
proto	cfapi	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .
port	9543, 9000, 6514 et 514	Port de communication que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs par défaut sont 9543 pour <code>cfapi</code> avec SSL activé, 9000 pour <code>cfapi</code> avec SSL désactivé, 6514 pour <code>syslog</code> avec SSL activé et 514 pour <code>syslog</code> avec SSL désactivé.
ssl	yes	Active ou désactive SSL. La valeur par défaut est oui. Lorsque <code>ssl</code> est défini sur oui, si vous ne définissez pas de valeur pour le port, le port 9543 est celui choisi automatiquement.

Paramètre	Valeur par défaut	Description
<code>max_disk_buffer</code>	200	Espace disque maximal utilisé par Log Insight Windows Agent (en Mo) pour mettre les événements et ses propres journaux en mémoire tampon. Lorsque la valeur maximale spécifiée pour <code>max_disk_buffer</code> est atteinte, l'agent commence à annuler les nouveaux événements entrants.
<code>debug_level</code>	0	Définit le niveau de détail des journaux. Reportez-vous à <a href="#">Définir le niveau de détail des journaux dans le Log Insight Agents</a> .

## Collecter les événements d'un fichier journal

Vous pouvez configurer l'agent vRealize Log Insight Linux afin de collecter des événements à partir d'un ou de plusieurs fichiers journaux.

Par défaut, l'agent Linux vRealize Log Insight collecte les fichiers cachés créés par des applications ou des éditeurs. Les noms des fichiers cachés commencent par un point. Pour empêcher l'agent Linux vRealize Log Insight de collecter les fichiers cachés, ajoutez le paramètre d'exclusion **`exclude=.*`**.

Les champs de noms sont limités. Les noms suivants sont réservés et ne peuvent pas être utilisés comme noms de champs.

- `event_type`
- `hostname`
- `source`
- `text`

Vous pouvez spécifier jusqu'à trois destinations pour les informations de l'agent et filtrer les informations avant leur envoi. Reportez-vous à [Transfert d'informations à partir d'un agent vRealize Log Insight](#)

**Note** La surveillance d'un grand nombre de fichiers, comme un millier ou plus, entraîne une utilisation plus élevée des ressources par vRealize Log Insight Agent et affecte les performances globales de la machine hôte. Pour éviter cela, configurez l'agent pour qu'il surveille uniquement les fichiers nécessaires à l'aide de modèles et de caractères génériques ou qu'il archive les anciens fichiers journaux. S'il n'est pas nécessaire de surveiller un grand nombre de fichiers, augmentez les paramètres de l'hôte, tels que le CPU et la RAM.

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.

- Vérifiez que l'agent Linux vRealize Log Insight est installé et en cours d'exécution. Connectez-vous à la machine Linux sur laquelle vous avez installé l'agent Linux vRealize Log Insight, ouvrez une console et exécutez `pgrep liagent`.

## Procédure

- 1 Ouvrez le fichier `/var/lib/loginsight-agent/liagent.ini` dans n'importe quel éditeur de texte.
- 2 Localisez la section `[server|<dest_id>]` du fichier. Ajoutez des paramètres de configuration et définissez les valeurs de votre environnement.

```
[filelog|section_name]
directory=path_to_log_directory
include=glob_pattern
...
```

Paramètre	Description
<code>[filelog section_name]</code>	Nom unique de la section de configuration.
<code>directory=full-path-to-log-file</code>	<p>Chemin d'accès complet au répertoire du fichier journal. Les modèles comportant des caractères génériques sont pris en charge. Exemples de configurations :</p> <ul style="list-style-type: none"> <li>■ Pour effectuer une collecte à partir de tous les sous-répertoires du répertoire <code>D:\Logs\new_test_logs</code>, utilisez <code>directory=D:\Logs\new_test_logs\*</code></li> <li>■ Si vos sous-répertoires disposent de leurs propres sous-répertoires, utilisez la configuration suivante pour surveiller tous les sous-répertoires <code>directory=D:\Logs\new_test_logs\*\*</code></li> </ul> <p><b>Note</b> Pour limiter le nombre de fichiers et de répertoires et éviter une consommation de ressources élevée, vous ne pouvez pas définir un répertoire glob pour les répertoires de premier ou de deuxième niveau, tels que <code>directory=c:/tmp/*</code> ou <code>directory=c:\Logs\*</code>. Le chemin au répertoire doit être d'au moins deux niveaux.</p> <p>Vous pouvez définir un chemin vers un répertoire non existant, et l'agent collectera les fichiers journaux dans ce répertoire une fois le répertoire et les fichiers créés.</p> <p>Vous pouvez définir le même répertoire pour une ou plusieurs sections de configuration différentes, ceci afin de collecter plusieurs fois les journaux depuis le même fichier. Ce processus rend possible l'application de différents filtres et différentes balises à la même source d'événements.</p> <p><b>Note</b> Si vous utilisez des configurations identiques pour ces sections, les événements en double apparaissent du côté serveur.</p>

Paramètre	Description
<code>include= file_name; ...</code>	<p>(Facultatif) Nom de fichier ou masque de fichier (modèle glob) à partir duquel collecter des données. Vous pouvez fournir des valeurs sous la forme d'une liste séparée par des points-virgules. La valeur par défaut est <code>*</code> qui prévoit l'inclusion de tous les fichiers. Le paramètre respecte la casse.</p> <p>Un masque de fichier (modèle glob) peut être utilisé pour grouper les fichiers qui suivent la même convention de dénomination, ainsi que dans un nom de fichier unique. Par exemple, les noms de fichier qui contiennent des espaces, tels que <code>vRealize Ops Analytics.log</code> et <code>vRealize Ops Collector.log</code>, peuvent être spécifiés avec <code>vRealize vRealize?Ops?Analytics*.log</code> ou <code>vRealize*.log</code>. En utilisant des masques de fichier, vous pouvez spécifier des noms de fichier qui sont acceptables pour la configuration de l'agent sous des hôtes Linux et Windows.</p> <p>Les fichiers <code>.zip</code> et <code>.gz</code> sont exclus par défaut de la collecte.</p> <p><b>Important</b> Si vous collectez un fichier journal archivé par rotation, utilisez les paramètres <code>include</code> et <code>exclude</code> pour spécifier un modèle générique correspondant au fichier principal et au fichier archivé par rotation. Si le modèle générique correspond uniquement au fichier journal principal, les agents vRealize Log Insight peuvent ne pas prendre en compte des événements lors de l'archivage par rotation. Les agents vRealize Log Insight déterminent automatiquement l'ordre correct des fichiers archivés par rotation et envoient les événements au serveur vRealize Log Insight dans l'ordre approprié. Par exemple, si votre fichier journal principal se nomme <code>myapp.log</code> et si les journaux archivés par rotation sont <code>myapp.log.1</code>, <code>myapp.log.2</code> etc., vous pouvez utiliser le modèle <code>include</code> suivant :</p> <pre>include= myapp.log;myapp.log.*</pre>
<code>exclude=regular_expression</code>	<p>(Facultatif) Nom ou masque de fichier (modèle générique) à exclure de la collecte. Vous pouvez fournir des valeurs sous la forme d'une liste séparée par des points-virgules. La liste est vide par défaut, ce qui signifie qu'aucun fichier n'est exclu.</p>
<code>event_marker=regular_expression</code>	<p>(Facultatif) Expression régulière qui désigne le début d'un événement dans le fichier journal. En cas d'omission, la valeur par défaut est nouvelle ligne. Les expressions que vous tapez doivent utiliser la syntaxe des expressions régulières Perl.</p> <p><b>Note</b> Les symboles, par exemple les guillemets (" "), ne sont pas traités comme des wrappers pour les expressions régulières. Ils sont traités comme partie intégrante du modèle.</p> <p>Étant donné que l'agent vRealize Log Insight est optimisé pour la collecte en temps réel, les messages journaux partiels écrits avec un délai interne peuvent être fractionnés en plusieurs événements. Si l'ajout de fichiers journaux s'arrête pendant plus de 200 ms sans nouvel <code>event_marker</code>, l'événement partiel est traité comme terminé, analysé et livré. Cette logique de minutage est non configurable et prioritaire par rapport au paramètre <code>event_marker</code>. Les appenders de fichiers journaux doivent purger les événements complets.</p>
<code>enabled=yes no</code>	<p>(Facultatif) Paramètre permettant d'activer ou de désactiver la section de configuration. Les valeurs possibles sont <code>yes</code> ou <code>no</code>. La valeur par défaut est <code>yes</code>.</p>

Paramètre	Description
<code>charset=char-encoding-type</code>	<p>(Facultatif) Codage de caractères des fichiers journaux que surveille l'agent. Les valeurs possibles sont :</p> <ul style="list-style-type: none"> <li>■ UTF-8</li> <li>■ UTF-16LE</li> <li>■ UTF-16BE</li> </ul> <p>La valeur par défaut est UTF-8.</p>
<code>tags={"tag-name": "tag-value", ...}</code>	<p>(Facultatif) Paramètre permettant d'ajouter des balises personnalisées aux champs d'événements collectés. Définissez des balises avec la notation JSON. Les noms de balises peuvent contenir des lettres, des chiffres et des traits de soulignement. Un nom de balise ne peut commencer que par une lettre ou un trait de soulignement, et ne doit pas dépasser 64 caractères. Les noms de balise ne sont pas sensibles à la casse. Par exemple, si vous utilisez <code>tags={"nom_balise1": "valeur balise 1", "Nom_Balise1": "valeur balise 2" }</code>, <code>Nom_Balise1</code> est ignoré, car il est en double. Vous ne pouvez pas utiliser <code>event_type</code> et <code>timestamp</code> comme noms de balises. Tous les doublons inclus dans la même déclaration sont ignorés.</p> <p>Si la destination est un serveur Syslog, les balises peuvent remplacer le champ APP-NAME. Par exemple, <code>tags={"appname": "VROPS"}</code>.</p>
<code>exclude_fields</code>	<p>(Facultatif) Paramètre permettant d'exclure des champs donnés d'une collecte. Vous pouvez fournir plusieurs valeurs sous forme de liste séparée par des points-virgules ou des virgules. Par exemple,</p> <ul style="list-style-type: none"> <li>■ <code>exclude_fields=hostname; filepath</code></li> <li>■ <code>exclude_fields=type; size</code></li> <li>■ <code>exclude_fields=type, size</code></li> </ul>
<code>raw_syslog=Yes No</code>	<p>Pour les agents utilisant le protocole Syslog, cette option autorise l'agent à collecter et à envoyer des événements Syslog bruts. La valeur par défaut est Non, ce qui signifie que les événements collectés sont transformés par des attributs Syslog spécifiés par l'utilisateur. Activez cette option pour collecter des événements sans transformations syslog.</p>

### 3 Enregistrez et fermez le fichier `liagent.ini`.

## Exemple : Configurations

```
[filelog|messages]
directory=/var/log
include=messages;messages.?

[filelog|syslog]
directory=/var/log
include=syslog;syslog.?

[filelog|Apache]
directory=/var/log/apache2
include=*
```

## Filtrer les événements

Vous pouvez filtrer tous les événements collectés sur l'agent Linux vRealize Log Insight selon leurs valeurs de champ afin de spécifier les événements de journal à sélectionner ou à annuler. Vous pouvez utiliser les options de collecteur `whitelist` et `blacklist` pour définir des filtres.

---

**Info-bulle** Par défaut, l'agent Linux vRealize Log Insight collecte les fichiers cachés créés par des programmes ou des éditeurs. Les noms de ces fichiers ne peuvent pas commencer par un point. Pour empêcher l'agent Linux vRealize Log Insight de collecter les fichiers cachés, ajoutez un paramètre `exclude=.*`.

---

Pour chaque événement, le collecteur évalue l'expression des filtres `whitelist` et `blacklist`. Si l'expression `whitelist` donne la valeur True et que l'expression `blacklist` donne la valeur False ou ne peut pas être évaluée, l'événement est déplacé vers la file d'attente en vue d'un traitement ultérieur. Dans tous les autres cas, le collecteur annule l'événement. La valeur par défaut de l'expression `whitelist` est True et celle de l'expression `blacklist` est False.

---

**Info-bulle** Le collecteur `Filelog` fournit moins de champs pour le filtrage. Pour obtenir des champs de filtrage, vous pouvez analyser les journaux. Pour plus d'informations, consultez [Analyse des journaux](#).

---

Un filtre `whitelist` ou `blacklist` est un ensemble de variables, de littéraux et d'opérateurs qui donne une valeur logique ou entière unique. Vous utilisez les champs d'événement comme des variables et les chaînes ainsi que les nombres entre guillemets doubles comme des littéraux. Pour plus d'informations sur les opérateurs que vous pouvez utiliser dans une expression de filtre, consultez la section [Champs d'événements et opérateurs](#).

---

### Note

- Si vous comparez un nombre avec une chaîne ou si la comparaison implique des chaînes numériques, chaque chaîne est convertie en nombre et la comparaison s'effectue numériquement. Par exemple :
  - L'expression `whitelist = 123.0 == "000123"` donne la valeur `True`.
  - L'expression `whitelist = "00987" == "987.00"` donne la valeur `True`.
  - Dans l'expression `whitelist = response_size >= "12.12"`, si le champ `response_size` contient une valeur numérique, l'expression est évaluée numériquement. Si la taille de la réponse est supérieure à 12,12, l'expression est `True`. Sinon, elle est `False`.
  - Dans l'expression `whitelist = "09123" < "234"`, les deux littéraux de chaîne sont convertis en valeurs numériques et l'expression donne la valeur `False`.
- Si l'un des opérandes de chaîne ne peut pas être converti en valeurs numériques, les deux opérandes sont convertis en chaîne. Une comparaison lexicographique simple sensible à la casse est effectuée. Par exemple :
  - L'expression `whitelist = "1234a" == "1234A"` est une comparaison de chaînes qui donne la valeur `False`.
  - L'expression `whitelist = 4 < "four"` convertit 4 en « 4 » et donne la valeur `True`.
  - Dans l'expression `whitelist = response_size > "thousand"`, la valeur du champ `response_size` est convertie en valeur de chaîne qui donne la valeur `False` à l'expression.
- Si une expression de filtre donne une valeur entière, elle est considérée comme `False` si elle est de 0 et comme `True` dans le cas contraire.

Par exemple, l'expression `whitelist = some_integer & 1` donne la valeur `True` si le champ `some_integer` a un bit de poids faible défini et `False` dans le cas contraire.

---

Pour accéder à la liste complète des champs d'événements et des opérateurs, consultez la section [Collecter les événements d'un fichier journal](#).

Dans cet exemple, vous collectez des journaux d'accès Apache depuis le fichier `/var/log/httpd/access`. Voici quelques exemples de journaux du fichier :

- `127.0.0.1 - frank [10/Oct/2016:13:55:36 +0400] "GET /apache_pb.gif HTTP/1.0" 200 2326`
- `198.51.100.56 - john [10/Oct/2016:14:15:31 +0400] "GET /some.gif HTTP/1.0" 200 8270`

- 198.51.100.12 - smith [10/Oct/2016:14:15:31 +0400] "GET /another.gif HTTP/1.0" 303 348
- 198.51.100.32 - test [10/Oct/2016:15:22:55 +0400] "GET /experimental\_page.gif HTTP/1.0" 400 46374
- 127.0.0.1 - test [10/Oct/2016:15:22:57 +0400] "GET /experimental\_page2.gif HTTP/1.0" 301 100

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé l'agent vRealize Log Insight Linux, ouvrez une console et exécutez `pgrep liagent` pour vérifier que l'agent vRealize Log Insight Linux est installé et fonctionne.

### Procédure

- 1 Définissez un analyseur pour les journaux, comme indiqué dans l'extrait de code suivant :

```
[parser|apache-access]
base_parser=clf
format=%h %l %u %t \"%r\" %s %b
```

L'analyseur que vous avez défini extrait les champs `remote_host`, `remote_log_name`, `remote_auth_user`, `timestamp`, `request`, `status_code` et `response_size` pour tous les événements collectés depuis le fichier `/var/log/httpd/access`. Vous pouvez utiliser ces champs pour filtrer les événements.

- 2 Ouvrez le fichier `/var/lib/loginsight-agent/liagent.ini` dans n'importe quel éditeur de texte.
- 3 Définissez une section `Filelog` dans le fichier pour collecter et analyser les journaux, comme indiqué dans l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
```

- 4 Filtrez les événements en fonction de vos besoins.
  - Pour collecter des journaux dont l'état de HTTP est 200, vous pouvez définir un élément `whitelist` dans la section `Filelog`, comme indiqué dans l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
whitelist = status_code == 200
```

L'expression `whitelist` donne la valeur `True` uniquement pour les premier et deuxième événements provenant des exemples de journaux. Ensuite, le collecteur sélectionne ces événements.

Si le champ `status_code` n'existe pas dans l'événement, car il n'est pas présent dans le journal ou n'est pas analysé, l'expression `whitelist` ne peut pas être évaluée. Cela signifie qu'elle donne la valeur `False` et que le collecteur annule l'événement.

- Pour annuler un événement qui ne vous intéresse pas, vous pouvez utiliser l'option `blacklist`. Par exemple, si vous n'êtes pas intéressé par le trafic local, vous pouvez bloquer l'adresse IP locale, comme indiqué dans l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
blacklist = remote_host == "127.0.0.1"
```

Le collecteur sélectionne les deuxième, troisième et quatrième événements provenant des exemples de journaux.

- Pour filtrer les événements en fonction de plusieurs prédicats, vous pouvez utiliser les opérateurs `or` et `and`. Par exemple, vous pouvez annuler des événements générés depuis une adresse IP locale ou des événements générés par des utilisateurs test depuis n'importe quel hôte non requis, comme indiqué dans l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
blacklist = remote_host == "127.0.0.1" or remote_auth_user == "test"
```

L'utilisation de l'opérateur `or` donne la valeur `True` à l'expression `blacklist` afin d'ignorer un événement indésirable. L'expression demande au collecteur d'annuler l'événement si la valeur de champ `remote_host` est « 127.0.0.1 » ou si la valeur de champ `remote_auth_user` est « test ».

Le collecteur sélectionne les deuxième et troisième événements provenant des exemples de journaux.

- Pour annuler des événements générés depuis une adresse IP locale par des utilisateurs test, vous pouvez utiliser `and` dans l'expression `blacklist`, comme indiqué dans l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
blacklist = remote_host == "127.0.0.1" and remote_auth_user == "test"
```

Le collecteur annule le cinquième événement provenant des exemples de journaux.

- Vous pouvez utiliser conjointement les filtres `whitelist` et `blacklist`. Par exemple, si vous avez besoin d'événements dont la taille de la réponse est supérieure à 1 024 octets, mais pas de ceux provenant d'un hôte local, vous pouvez utiliser l'extrait de code suivant :

```
[filelog|apache-access]
directory = /var/log/httpd/
include = access
parser = apache-access
whitelist = response_size > 1024
blacklist = remote_host == "127.0.0.1" or remote_host == "localhost"
```

Le collecteur sélectionne le deuxième événement provenant des exemples de journaux.

## Collecte d'événements depuis `journald`

À partir de vRealize Log Insight 4.6, les agents peuvent lire des journaux du service de système `journald` pour les données de journal dans les distributions Linux exécutant `systemd`. `journald` est désormais la norme par défaut pour la journalisation dans les plates-formes Linux basées sur `systemd`. La section de configuration `journald` prend en charge les options suivantes :

### `journal_files`

Fichiers journaux à surveiller. Les valeurs suivantes sont prises en charge :

Valeur	Description
all	Ouvrez et surveillez tous les fichiers journaux disponibles.
local	Surveillez et lisez uniquement les fichiers journaux générés sur la machine locale.
runtime	Surveillez et lisez uniquement les fichiers journaux volatils, en excluant les fichiers dans le stockage persistant.
system	Surveillez et lisez uniquement les services système et les fichiers journaux du noyau.
user	Surveillez et lisez uniquement les fichiers journaux de l'utilisateur actuel.

### `fetch_fields`

Champs à extraire avec le message des entrées du journal. La valeur de cette option est une liste de noms de champs insensible à la casse séparés par des virgules. Les valeurs suivantes sont prises en charge :

Valeur	Description
pri_severity,pri_facility,syslog_identifiant	Valeur par défaut pour cette option.
*	Extrayez tous les champs.
all	N'extrayez pas de champs.

# Filtrage des événements provenant des agents vRealize Log Insight

Vous pouvez fournir les informations qu'un agent envoie à une destination avec l'option de filtre dans la section `[server|<dest_id>]` du fichier local `liagent.ini`.

L'option est au format suivant :

```
filter = {collector_type; collector_filter; event_filter}
```

Type de filtre	Description
<code>collector_type</code>	Liste séparée par des virgules qui définit les types de collecteur. Les valeurs prises en charge sont <code>filelog</code> ou <code>winlog</code> . Si aucune valeur n'est fournie, tous les types de collecteurs sont utilisés.
<code>collector_filter</code>	Spécifie le nom d'une section de collecteur au format regex. Par exemple, <code>vcops_.*</code> fait référence à toutes les sections de collecteur qui commencent par « <code>vcops_</code> ».
<code>event_filter</code>	Les filtres des champs d'événements utilisent la même syntaxe que les listes autorisées ou bloquées dans les sections de collecteur. Un agent n'envoie que les événements avec l'expression <code>True</code> ou une valeur non nulle. Un <code>event_filter</code> vide a toujours la valeur <code>True</code> . Pour utiliser <code>event_filter</code> sur des événements, vous devez définir un analyseur dans les sections de collecteur appropriées pour l'extraction de champs. Si une expression ne peut pas être évaluée en raison de l'absence de champs dans l'événement collecté, celui-ci est abandonné.

Vous pouvez spécifier plusieurs expressions de filtre en les séparant par une virgule, comme indiqué dans l'exemple suivant :

```
filter={winlog;Micr.*;},{filelog;apache-access;level=="error"}
```

Si un message répond à plusieurs critères de filtre pour une cible de destination, il n'est envoyé qu'une seule fois.

**Tableau 4-1. Exemples de syntaxe**

Filtre	Signification
<code>filter= {winlog;Microsoft.*;}</code>	Envoie des événements à partir de collecteurs <code>winlog</code> uniquement si le nom de l'événement commence par « <code>Microsoft</code> ».
<code>filter= {winlog;Microsoft.*; eventid == 1023}</code>	Envoie des événements à partir de collecteurs <code>winlog</code> uniquement si le nom de l'événement commence par « <code>Microsoft</code> » et si l'ID d'événement est égal à <code>1023</code> .
<code>filter= {.*;}</code>	Valeur de filtre par défaut. Envoie tous les événements depuis toutes les sources.
<code>filter= {winlog;.*;}</code>	Envoie tous les événements depuis des sections <code>winlog</code> .
<code>filter= {filelog;syslog;facility&lt;5}</code>	Envoie des événements à partir de la section <code>[filelog syslog]</code> si les installations sont inférieures à <code>5</code> . Les sections <code>[filelog syslog]</code> doivent disposer d'un analyseur qui extrait le champ d'installation. Sinon, tous les événements sont ignorés.
<code>filter= {;;}</code>	Ne correspond à aucun événement. Utilisez cette syntaxe pour désactiver le transfert d'événements.

L'exemple suivant ajoute un filtre à la configuration de la deuxième destination de l'exemple précédent.

```
; The second destination receives just syslog events through the plain syslog protocol.
[server|syslog-audit]
hostname=third_party_audit_management.eng.vmware.com
proto=syslog
ssl=no
filter= {filelog; syslog; }
```

L'exemple suivant utilise une expression de filtre plus complexe.

```
; This destination receives vRealize Operations Manager events if they have the level field
equal
;to "error" or "warning" and they are collected by sections whose name begins with "vrops-"

[server|licf-prod1]
hostname=vrops-errors.licf.vmware.com
filter= {; vrops-.*; level == "error" || level == "warning"}
```

Vous pouvez spécifier plusieurs expressions de filtre en les séparant par une virgule, comme indiqué dans l'exemple suivant.

```
filter= e.
{winlog;Micr.*;},{filelog;apache-access;level=="error"}
```

## Transfert d'informations à partir d'un agent vRealize Log Insight

Vous pouvez transférer des événements collectés par un agent vers trois destinations au maximum. Une destination peut inclure des serveurs ou un redirecteur vRealize Log Insight, ou bien des solutions tierces de gestion des journaux.

Par exemple, vous voulez peut-être envoyer des journaux d'audit ou de système à un serveur pour votre équipe de sécurité, des journaux d'application à un serveur d'équipe DevOps et des journaux de mesures à un système de gestion informatique. Vous utilisez des filtres pour spécifier les informations qui sont envoyées à une destination. Vous pouvez transférer des informations à partir d'un seul agent vRealize Log Insight vers trois destinations au maximum.

La configuration de l'agent est effectuée dans la section `[server|<dest_id>]` de votre fichier `liagent.ini` local. Utilisez le protocole `cfapi` avec des serveurs ou des redirecteurs vRealize Log Insight et `Syslog` avec d'autres cibles ou destinations.

Lorsque vous spécifiez plusieurs destinations pour un agent, la première utilise l'emplacement `loginsight` par défaut. Vous devez spécifier des informations sur l'emplacement pour les autres destinations.

L'exemple suivant montre une partie d'un fichier `liagent.ini` qui spécifie deux destinations. Le nom du serveur par défaut `loginsight` est implicite pour la première destination par défaut et n'est pas spécifié. La deuxième section `[server|<dest_id>]` spécifie une destination.

```
; The first (default) destination receives all collected events.
[server]
ssl=yes

; The second destination receives just syslog events through the plain syslog protocol.
[server|syslog-audit]
hostname=third_party_audit_management.eng.vmware.com
proto=syslog
ssl=no
```

Pour plus d'informations sur la création de filtres pour les agents, reportez-vous à la section [Filtrage des événements provenant des agents vRealize Log Insight](#).

## Définir le serveur vRealize Log Insight cible

Vous pouvez définir ou modifier le serveur vRealize Log Insight cible pour un agent vRealize Log Insight exécuté sur Windows. Vous pouvez envoyer des événements à trois destinations au maximum et filtrer la sortie par destination.

La destination par défaut peut être configurée dans la section `[server]` du fichier `liagent.ini`. La destination par défaut est toujours présente et, par défaut, le nom d'hôte est défini sur `loginsight`. Pour ajouter davantage de destinations cibles, créez une section `[server|<dest_id>]` pour chaque cible. Vous devez spécifier un nom d'hôte unique comme ID de destination pour chaque connexion supplémentaire. Pour les destinations supplémentaires, vous pouvez utiliser les mêmes options que celles de la section `[server]` par défaut. Ne configurez pas de destinations supplémentaires pour la mise à niveau automatique ou ne les utilisez pas pour la configuration de l'agent. Vous pouvez indiquer deux destinations supplémentaires.

Par défaut, l'agent envoie tous les événements collectés à toutes les destinations. Vous pouvez filtrer les événements afin d'envoyer des événements différents vers des destinations différentes avec l'option `file`. Pour plus d'informations, consultez [Filtrage des événements provenant des agents vRealize Log Insight](#).

### Conditions préalables

- Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.
- Si vous disposez d'un cluster vRealize Log Insight pour lequel ILB (Integrated Load Balancer) est activé, consultez [Activer l'équilibrage de charge intégré](#) pour connaître les exigences spécifiques d'un certificat SSL personnalisé.

## Procédure

- 1 Accédez au répertoire des données du programme de l'agent vRealize Log Insight Windows.  
%ProgramData%\VMware\Log Insight Agent
- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.
- 3 Modifiez les paramètres suivants et définissez les valeurs correspondant à votre environnement.

Paramètre	Description
<b>proto</b>	<p>Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code>. La valeur par défaut est <code>cfapi</code>.</p>
<b>hostname</b>	<p>Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. Vous pouvez spécifier une adresse IPv4 ou IPv6. Une adresse IPv6 peut être spécifiée avec ou sans crochets. Par exemple :</p> <pre>hostname = 2001:cdba::3257:9652 or hostname = [2001:cdba::3257:9652]</pre> <p>Si l'hôte prend en charge les piles IPv4 et IPv6, et qu'un nom de domaine est spécifié comme nom d'hôte, l'agent choisit la pile IP en fonction de l'adresse IP renvoyée par le programme de résolution de noms. Si le programme de résolution renvoie des adresses IPv4 et IPv6, l'agent tente de se connecter séquentiellement aux deux adresses dans l'ordre donné.</p>
<b>max_disk_buffer</b>	<p>Espace disque maximal en Mo que l'agent Windows Log Insight peut utiliser pour mettre en mémoire tampon les événements collectés pour ce serveur particulier. L'option remplace la valeur <code>[storage].max_disk_buffer</code> pour ce serveur. La valeur par défaut est de 150 Mo et vous pouvez définir le tampon sur une taille comprise entre 50 et 8 000 Mo.</p>
<b>port</b>	<p>Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut.</p> <ul style="list-style-type: none"> <li>■ cfapi avec SSL activé : 9543</li> <li>■ cfapi avec SSL désactivé : 9000</li> <li>■ syslog avec SSL activé : 6514</li> <li>■ syslog avec SSL désactivé : 514</li> </ul>
<b>ssl</b>	<p>Active ou désactive SSL. La valeur par défaut est oui. Lorsque <code>ssl</code> est défini sur oui, le port est défini sur 9543, sauf spécification contraire.</p>

Paramètre	Description
<b>reconnect</b>	Temps en minutes pour forcer la reconnexion au serveur. La valeur par défaut est 30.
<b>filter</b>	Spécifie les informations qu'un agent envoie à une destination. Cette option utilise trois arguments :  <code>{collector_type; collector_filter; event_filter}</code>

```
[server]
hostname=LOGINSIGHT
; Hostname or IP address of your Log Insight server / cluster load balancer. Default:
;hostname=LOGINSIGHT

; Protocol can be cfapi (Log Insight REST API), syslog. Default:
;proto=cfapi

; Log Insight server port to connect to. Default ports for protocols (all TCP):
; syslog: 514; syslog with ssl: 6514; cfapi: 9000; cfapi with ssl: 9543. Default:
;port=9543

; SSL usage. Default:
;ssl=yes
```

#### 4 Enregistrez et fermez le fichier `liagent.ini`.

##### Exemple

L'exemple de configuration suivant définit un serveur vRealize Log Insight cible qui utilise une autorité de certification approuvée.

```
[server]
proto=cfapi
hostname=LOGINSIGHT
port=9543
ssl=yes;
ssl_ca_path=/etc/pki/tls/certs/ca.pem
```

L'exemple suivant montre une configuration à plusieurs destinations qui inclut le filtrage des messages par destination.

```
; The first (default) destination receives all collected events.
[server]
hostname=prod1.licf.vmware.com

; The second destination receives just syslog events through the plain syslog protocol.
[server|syslog-audit]
hostname=third_party_audit_management.eng.vmware.com
proto=syslog
ssl=no
filter={filelog; syslog; }
```

```

; The third destination receives vRealize Operations Manager events if they have the level
field equal to "error" or "warning"
; and they are collected by sections whose name begins with "vrops-"

[server|licf-prod1]
hostname=vrops-errors.licf.vmware.com
filter={; vrops-.*; level == "error" || level == "warning"}

; Collecting syslog messages.
[filelog|syslog]
directory=/var/log
include=messages

; various vROPs logs. Note that all section names begin with a "vrops-" prefix, which is used
in third destination filter.
[filelog|vrops-ANALYTICS-analytics]
directory=/data/vcops/log
include=analytics*.log*
exclude=analytics*-gc.log*
parser=auto

[filelog|vrops-COLLECTOR-collector]
directory=/data/vcops/log
include=collector.log*
event_marker=^{\d{4}-\d{2}-\d{2}} [\s]\d{2}:\d{2}:\d{2}\.\d{3}
parser=auto

[filelog|vrops-COLLECTOR-collector_wrapper]
directory=/data/vcops/log
include=collector-wrapper.log*
event_marker=^{\d{4}-\d{2}-\d{2}} [\s]\d{2}:\d{2}:\d{2}\.\d{3}
parser=auto

```

### Étape suivante

Vous pouvez configurer des options SSL supplémentaires pour l'agent vRealize Log Insight. Reportez-vous à la rubrique [Configurer une connexion SSL entre le serveur et les Log Insight Agents](#).

## Spécifier la cible d'un agent

Vous pouvez spécifier jusqu'à trois destinations auxquelles l'agent Linux vRealize Log Insight va envoyer des événements.

La section `[server|<dest_id>]` du fichier `li-agent.ini` permet de définir des connexions de destinations multiples. La variable `<dest_id>` qu'elle contient est un ID de connexion unique par configuration. Pour les destinations supplémentaires, vous pouvez utiliser les mêmes options que celles de la section `[server]` par défaut. Toutefois, ne configurez pas de destinations supplémentaires pour la mise à niveau automatique ou ne les utilisez pas pour la configuration de l'agent. Vous pouvez indiquer deux destinations supplémentaires.

La première cible que vous définissez peut utiliser la valeur par défaut du serveur `loginsight`. Lorsque vous définissez des cibles supplémentaires, vous devez spécifier un nom d'hôte dans les sections `[server]` des cibles suivantes. Sans le filtrage, l'agent envoie tous les événements collectés à toutes les destinations. Il s'agit de la configuration par défaut. Toutefois, vous pouvez filtrer les événements afin d'envoyer des événements différents vers des destinations différentes.

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé l'agent vRealize Log Insight Linux, ouvrez une console et exécutez `pgrep liagent` pour vérifier que l'agent vRealize Log Insight Linux est installé et fonctionne.
- Si vous disposez d'un cluster vRealize Log Insight pour lequel ILB (Integrated Load Balancer) est activé, consultez [Activer l'équilibrage de charge intégré](#) pour connaître les exigences spécifiques d'un certificat SSL personnalisé.

### Procédure

- 1 Ouvrez le fichier `/var/lib/loginsight-agent/liagent.ini` dans n'importe quel éditeur de texte.
- 2 Modifiez les paramètres suivants et définissez les valeurs correspondant à votre environnement.

Paramètre	Description
<code>proto</code>	Protocole que l'agent utilise pour envoyer des événements au serveur vRealize Log Insight. Les valeurs possibles sont <code>cfapi</code> et <code>syslog</code> . La valeur par défaut est <code>cfapi</code> .
<code>hostname</code>	Adresse IP ou nom d'hôte du dispositif virtuel vRealize Log Insight. Vous pouvez spécifier une adresse IPv4 ou IPv6. Une adresse IPv6 peut être spécifiée avec ou sans crochets. Par exemple : <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <pre>hostname = 2001:cdba::3257:9652 or hostname = [2001:cdba::3257:9652]</pre> </div> Si l'hôte prend en charge les piles IPv4 et IPv6, et qu'un nom de domaine est spécifié comme nom d'hôte, l'agent utilise la pile IP en fonction de l'adresse IP qui est renvoyée par le programme de résolution de noms. Si le programme de résolution renvoie des adresses IPv4 et IPv6, l'agent tente de se connecter séquentiellement aux deux adresses dans l'ordre donné.
<code>max_disk_buffer</code>	Espace disque maximal en Mo que l'agent Linux Log Insight peut utiliser pour mettre en mémoire tampon les événements collectés pour ce serveur particulier. L'option remplace la valeur <code>[storage].max_disk_buffer</code> pour ce serveur. La valeur par défaut est de 150 Mo et vous pouvez définir le tampon sur une taille comprise entre 50 et 8 000 Mo.

Paramètre	Description
<b>port</b>	Port de communication utilisé par l'agent pour envoyer des événements au serveur vRealize Log Insight ou tiers. Par défaut, l'agent utilise le port adéquat en fonction des options qui sont définies pour SSL et le protocole. Consultez les valeurs de port par défaut mentionnées dans la liste ci-dessous. Vous devez uniquement spécifier l'option de port si elle est différente de ces valeurs par défaut. <ul style="list-style-type: none"> <li>■ cfapi avec SSL activé : 9543</li> <li>■ cfapi avec SSL désactivé : 9000</li> <li>■ syslog avec SSL activé : 6514</li> <li>■ syslog avec SSL désactivé : 514</li> </ul>
<b>ssl</b>	Active ou désactive SSL. La valeur par défaut est oui. Lorsque <code>ssl</code> est défini sur oui, si vous ne définissez pas de valeur pour le port, le port 9543 est celui choisi automatiquement.
<b>reconnect</b>	Durée en minutes pour forcer la reconnexion au serveur. La valeur par défaut est 30.

```
[server]
hostname=LOGINSIGHT
; Hostname or IP address of your Log Insight server / cluster load balancer. Default:
;hostname=LOGINSIGHT

; Protocol can be cfapi (Log Insight REST API), syslog. Default:
;proto=cfapi

; Log Insight server port to connect to. Default ports for protocols (all TCP):
; syslog: 514; syslog with ssl: 6514; cfapi: 9000; cfapi with ssl: 9543. Default:
;port=9543

; SSL usage. Default:
;ssl=yes
```

### 3 Enregistrez et fermez le fichier `liagent.ini`.

#### Exemple

L'exemple de configuration suivant définit un serveur vRealize Log Insight cible qui utilise une autorité de certification approuvée.

```
[server]
proto=cfapi
hostname=LOGINSIGHT
port=9543
ssl=yes;
ssl_ca_path=/etc/pki/tls/certs/ca.pem
```

L'exemple suivant illustre une configuration de destinations multiples.

- La première destination (par défaut) reçoit tous les événements collectés.

```
[server]
hostname=prod1.licf.vmware.com
```

- La deuxième destination reçoit uniquement les événements syslog via le protocole syslog normal.

```
[server|syslog-audit]
hostname=third_party_audit_management.eng.vmware.com
proto=syslog
ssl=no
filter= {filelog; syslog; }
```

- La troisième destination reçoit les événements vRealize Operations Manager à condition qu'ils comportent un champ de niveau contenant « erreur » ou « avertissement » et qu'ils aient été collectés par des sections dont le nom commence par « vrops- »

```
[server|licf-prod1]
hostname=vrops-errors.licf.vmware.com
filter= {; vrops-.*; level == "error" || level == "warning"}

;Collecting syslog messages.
[filelog|syslog]
directory=/var/log
include=messages

;various vRops logs. Note that all section names begin with "vrops-" prefix, which is used in
third destination filter.
[filelog|vrops-ANALYTICS-analytics]
directory=/data/vcops/log
include=analytics*.log*
exclude=analytics*-gc.log*
parser=auto
[filelog|vrops-COLLECTOR-collector]
directory=/data/vcops/log
include=collector.log*
event_marker=^\\d
{4}-\\d{2}-\\d{2} [\\s]\\d{2}:\\d{2}:\\d{2} \\, \\d{3}
parser=auto

[filelog|vrops-COLLECTOR-collector_wrapper]
directory=/data/vcops/log
include=collector-wrapper.log*
event_marker=^\\d{4}
-\\d
{2}-\\d{2}
[\\s]\\d
{2}:\\d{2}
:\\d
{2}
\\.\\d
```

```
{3}  
parser=auto
```

### Étape suivante

Vous pouvez configurer des options SSL supplémentaires pour l'agent vRealize Log Insight Linux. Reportez-vous à la rubrique [Configurer une connexion SSL entre le serveur et les Log Insight Agents](#).

## Configuration centralisée des agents vRealize Log Insight

Vous pouvez configurer plusieurs agents vRealize Log Insight.

Chaque agent vRealize Log Insight dispose d'une configuration locale et d'une configuration côté serveur. La configuration locale est stockée dans le fichier `liagent.ini` sur la machine virtuelle ou physique sur laquelle l'agent vRealize Log Insight est installé. La configuration côté serveur est accessible et modifiable, par exemple, depuis **Administration > Agents** dans l'interface utilisateur Web. La configuration de chaque agent vRealize Log Insight se compose de sections et de clés. Les clés possèdent des valeurs configurables.

Les agents vRealize Log Insight interrogent régulièrement le serveur vRealize Log Insight et reçoivent la configuration côté serveur. La configuration côté serveur et la configuration locale sont fusionnées pour former la configuration effective. La configuration effective est utilisée comme configuration opérationnelle par chaque agent vRealize Log Insight. Les configurations sont fusionnées section par section et clé par clé. Les valeurs de la configuration côté serveur remplacent les valeurs de la configuration locale. Les règles de fusion sont les suivantes :

- Si une section est présente uniquement dans la configuration locale ou uniquement dans la configuration côté serveur, cette section et tout son contenu sont inclus à la configuration effective.
- Si une section est présente dans la configuration locale et dans la configuration côté serveur, les clés de cette section sont fusionnées selon les règles suivantes :
  - Si une clé est présente uniquement dans la configuration locale ou uniquement dans la configuration côté serveur, cette clé et sa valeur sont incluses à cette section dans la configuration effective.
  - Si une clé est présente dans la configuration locale et dans la configuration côté serveur, elle est incluse à cette section dans la configuration effective, avec la valeur qui lui est attribuée dans la configuration côté serveur.

Un utilisateur Admin vRealize Log Insight peut appliquer une configuration centralisée à tous les agents vRealize Log Insight. Par exemple, accédez à la page **Administration**, puis dans la section **Gestion**, cliquez sur **Agents**. Entrez les paramètres de configuration dans la zone **Configuration de l'agent**, puis cliquez sur **Enregistrer la configuration pour tous les agents**. La configuration s'applique à tous les agents actifs configurables au cours du cycle d'interrogation suivant.

Un utilisateur Admin vRealize Log Insight peut utiliser des filtres spécifiques dans des groupes d'agents, tels que le système d'exploitation, la version de l'agent, le nom d'hôte ou les plages d'adresses IP, et appliquer la configuration à des agents vRealize Log Insight spécifiques. Pour plus d'informations sur les groupes d'agents, consultez la section *Utilisation de groupes d'agents*.

### Note

- Vous ne pouvez appliquer une configuration centralisée qu'aux agents vRealize Log Insight qui utilisent le protocole cfapi.
- Un agent vRealize Log Insight n'est pas configurable dans l'un des scénarios suivants :
  - Le serveur vRealize Log Insight actuel n'est pas une destination principale. Pour plus d'informations sur la configuration de plusieurs destinations, consultez la section [Spécifier la cible d'un agent](#) .
  - Le paramètre `central_config = no` est utilisé dans la configuration de l'agent. Pour plus d'informations sur la configuration par défaut de l'agent pour Windows, consultez la section [Configuration par défaut de Log Insight Windows Agent](#).

## Exemple de fusion de configurations

Exemple de fusion de configurations locale et côté serveur de Log Insight Windows Agent.

### Configuration locale

Vous pouvez avoir la configuration locale suivante de Log Insight Windows Agent.

```
[server]
proto=cfapi
hostname=HOST
port=9000

[winlog|Application]
channel=Application

[winlog|Security]
channel=Security

[winlog|System]
channel=System

[filelog|ApacheAccessLogs]
enabled=yes
directory=C:\Program Files (x86)\Apache Software Foundation\Apache2.2\logs
include=*.log
exclude=*_old.log
event_marker=^(\d{1,3}\.){3}\d{1,3} - -
```

## Configuration côté serveur

Vous pouvez utiliser la page **Administration > Agents** de l'interface utilisateur Web pour appliquer la configuration centralisée à tous les agents. Par exemple, vous pouvez exclure et ajouter des canaux de collecte et modifier le paramètre de reconnexion par défaut.

```
[server]
reconnect=20

[winlog|Security]
channel=Security
enabled=no

[winlog|Microsoft-Windows-DeviceSetupManagerOperational]
channel=Microsoft-Windows-DeviceSetupManager/Operational
```

## Configuration valide

La configuration valide est le résultat de la fusion des configurations locale et côté serveur. Log Insight Windows Agent est configuré pour :

- se reconnecter au serveur vRealize Log Insight toutes les 20 minutes
- continuer de collecter les canaux d'événements Application et Système
- arrêter la collecte du canal d'événements Sécurité
- démarrer la collecte du canal d'événements Microsoft-Windows-DeviceSetupManager/Operational
- continuer de collecter les ApacheAccessLogs

```
[server]
proto=cfapi
hostname=HOST
port=9000
reconnect=20

[winlog|Application]
channel=Application

[winlog|Security]
channel=Security
enabled=no

[winlog|System]
channel=System

[winlog|Microsoft-Windows-DeviceSetupManagerOperational]
channel=Microsoft-Windows-DeviceSetupManager/Operational

[filelog|ApacheAccessLogs]
enabled=yes
directory=C:\Program Files (x86)\Apache Software Foundation\Apache2.2\logs
include=*.log
```

```
exclude=*_old.log
event_marker=^(\d{1,3}\.){3}\d{1,3} - -
```

## Utilisation de valeurs communes pour la configuration des agents

Vous pouvez remplacer les valeurs par défaut du fichier de configuration de l'agent par des valeurs de paramètre communes qui s'appliquent à la section de configuration de chaque agent pour les agents Windows ou Linux.

### Options communes

Les options spécifiées dans la section `[common|global]` du fichier de configuration `liagent.ini` sont propagées sur toutes les sections, les options spécifiées dans la section `[common|filelog]` sont propagées uniquement sur les sections `filelog`, et les options `[common|winlog]` sont propagées uniquement sur les sections `winlog`.

Vous pouvez définir les paramètres suivants dans les sections communes : `tags`, `include`, `exclude`, `event_marker`, `charset`, `exclude_fields` et `parser` comme indiqué dans l'exemple ci-dessous. Cet exemple s'applique à un agent Windows :

```
[common|global]

tags = {"log_source_vm":"win-2008r2-64"}
exclude_fields = test_tag;some_other_tag
parser = auto

[common|filelog]
tags = {"collector_type":"filelog"}
exclude = *.trc

[filelog|channel_1]
directory = C:\app\log
include = *.log

...
```

Cet exemple spécifie le comportement suivant :

- Tous les journaux des sections `filelog` contiennent des balises `log_source_vm` et `collector_type` avec leurs valeurs correspondantes.
- Les balises `test_tag` et `some_other_tag` sont exclues de tous les journaux envoyés.
- L'analyseur `auto` est appliqué à tous les journaux collectés.
- Par défaut, tous les collecteurs `filelog` excluent les fichiers `*.trc` de la surveillance.

Les options dans `[common|global]` s'appliquent également à toutes les sections `winlog`.

## Fusionner et remplacer des critères

Si des options sont définies dans plusieurs sections, leurs valeurs sont fusionnées ou remplacées et la section avec une étendue inférieure a une priorité supérieure lors de la fusion/du remplacement. Cela signifie qu'une valeur de `[common|global]` est fusionnée avec ou remplacée par une valeur de `[common|filelog]`, qui en retour est combinée avec ou remplacée par une valeur de `[filelog|sample_section]`.

Le comportement Fusionner et remplacer suit les règles suivantes :

- Les options dont les valeurs représentent une liste de valeurs (tags, include, exclude et exclude\_fields) sont fusionnées avec des valeurs de cette option d'une section avec une priorité supérieure. Et, dans le cas de balises, les valeurs tags de sections avec une priorité supérieures remplacent la valeur de cette même balise d'une section avec une priorité inférieure, comme décrit précédemment.
- Les valeurs d'options pouvant avoir une valeur unique (event\_marker, charset et parser) sont remplacées par des valeurs de cette option de sections avec une priorité supérieure.

Cela signifie que la valeur de charset=UTF-8 de `[filelog|sample_section]` remplace la valeur générale de charset= UTF-16LE de `[common|global]`.

Par exemple, si vous disposez de `tags={"app":"global-test"}` dans `[common|filelog]` et de `tags={"app":"local-test","section":"flg_test_section"}` dans `[filelog|flg_test_section]`, la valeur de la balise « app » de la section `[filelog|flg_test_section]` remplace la valeur de `[common|filelog]`. Tous les journaux collectés via cette section filelog auront une balise « app » supplémentaire avec la valeur « local-test » et une balise « section » avec la valeur « flg\_test\_section ». Pour les sections winlog, la chaîne de priorité est la même, avec n'importe quelle section `[winlog|...]` avec la priorité la plus élevée et `[common|global]` avec la priorité la plus faible.

Lorsque des valeurs non valides sont spécifiées dans des sections communes, elles sont en général ignorées et non fusionnées avec des valeurs de sections filelog/winlog précédentes et correspondantes. Dans le cas de valeurs non valides dans des balises ou des options `exclude_fields`, l'agent extrait autant de données valides que possible et ignore le reste du fichier lorsque des données non valides sont détectées. Toutes les anomalies sont signalées dans le fichier journal de l'agent. Consultez le fichier journal si un comportement inattendu est rencontré et corrigez toutes les erreurs signalées par l'agent.

Si l'agent détecte une valeur non valide pour une option dans une section filelog ou winlog, il ne fusionne pas les valeurs d'option de cette section avec les valeurs d'option de sections communes et il n'active pas cette section. Toutes les erreurs sont signalées dans un fichier journal de l'agent. Consultez le fichier journal si un comportement inattendu est rencontré et corrigez toutes les erreurs signalées par l'agent.

## Analyse des journaux

Les analyseurs de journaux provenant d'agents extraient les données structurées de journaux bruts avant de les livrer au serveur vRealize Log Insight. Grâce à ces analyseurs, vRealize Log Insight peut analyser des journaux, en extraire des informations et afficher ces résultats sur le serveur. Vous pouvez configurer les analyseurs de journaux pour les agents vRealize Log Insight

Si le protocole syslog est utilisé, les champs extraits par les analyseurs font partie de STRUCTURED-DATA conformément à RFC5424.

## Configurer des analyseurs de journaux

Vous pouvez configurer les analyseurs pour les collecteurs FileLog et WinLog.

### Conditions préalables

Pour l'agent vRealize Log Insight Linux :

- Connectez-vous en tant que `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console et exécutez `pgrep liagent` pour vérifier que le Log Insight Linux Agent est installé et qu'il fonctionne.

Pour l'agent vRealize Log Insight Windows :

- Connectez-vous à la machine Windows sur laquelle vous avez installé Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight est installé.

### Procédure

- 1 Accédez au dossier contenant le fichier `liagent.ini`.

Système d'exploitation	Chemin
Linux	<code>/var/lib/loginsight-agent/</code>
Windows	<code>%ProgramData%\VMware\Log Insight Agent</code>

- 2 Ouvrez le fichier `liagent.ini` dans l'éditeur de texte de votre choix.
- 3 Pour configurer un analyseur, définissez une section d'analyseur. `[parser|myparser]`  
`myparser` est un nom arbitraire de l'analyseur qui peut être issu des sources du journal. La section d'analyseur doit faire référence à un analyseur intégré (ou à tout autre analyseur défini) et configurer les options obligatoires de cet analyseur ainsi que les options facultatives, si nécessaire.

Par exemple, `base_parser=csv` montre que l'analyseur `myparser` est dérivé de la commande `csv` de l'analyseur intégré. Pour cet analyseur, les journaux d'entrées doivent comprendre deux champs séparés par un point-virgule.

```
[parser|myparser]

base_parser=csv

fields=field_name1,field_name2

delimiter=";"
```

- 4 Après avoir défini `myparser`, appelez-le selon les sources de journaux `winlog` ou `filelog`.

```
[filelog|some_csv_logs]

directory=D:\Logs

include=*.txt;*.txt.*

parser=myparser
```

Les journaux collectés à partir des sources `some_csv_logs`, par exemple à partir du répertoire `D:\Logs`, sont analysés par `myparser` et les événements extraits s'affichent sur le serveur comme `field_name1` et `field_name2`, respectivement.

---

**Note** Les journaux statiques du répertoire `D:\Logs` ne sont pas entraînés dans le vRealize Log Insight par l'agent. Cependant, les nouveaux fichiers créés dans le répertoire `D:\Logs` sont disponibles dans vRealize Log Insight.

---

- 5 Enregistrez et fermez le fichier `liagent.ini`.

## Options communes aux analyseurs

Vous pouvez configurer des options communes à tous les analyseurs qui produisent des champs nommés.

### Mots réservés pour les noms de champs

Les champs de noms sont limités. Les noms suivants sont réservés et ne peuvent pas être utilisés comme noms de champs.

- `event_type`
- `hostname`
- `source`
- `text`

### Options communes aux analyseurs

Les options dans le tableau suivant peuvent être utilisées avec tous les analyseurs pris en charge.

Option	Description
<code>base_parser</code>	Le nom de l'analyseur de base étendu par cet analyseur personnalisé. Il peut s'agir d'un nom d'analyseur intégré ou d'un autre nom d'analyseur de client. Cette clé de configuration est obligatoire.
<code>field_decoder</code>	<p>Analyseurs imbriqués spécifiés en tant que chaînes JSON. Les clés représentent les noms des champs auxquels vous devez appliquer l'analyseur imbriqué et la valeur correspond au nom de l'analyseur à utiliser pour ce champ. Chaque analyseur imbriqué est appliqué au champ approprié décodé par l'analyseur de base. Les décodeurs de champ sont utiles lorsque la valeur d'un champ est une valeur complexe, par exemple un horodatage. L'option <b>field_decoder</b> prend également en charge des objets JSON plus complexes en tant qu'arguments qui vous permettent d'utiliser des conditions pour des valeurs de champ spécifiques qui sont vérifiées avant l'application de l'analyseur imbriqué.</p> <p><b>Note</b> Pour plus d'informations sur l'utilisation et les configurations conditionnelles, reportez-vous à la section Configurations conditionnelles de l'option <code>field_decoder</code> ci-dessous.</p>
<code>field_rename</code>	Cette option renomme les champs extraits. Utilisez une chaîne JSON dans laquelle les clés sont les noms d'origine des champs, et les valeurs sont les nouveaux noms des champs. L'option <code>field_decoder</code> est toujours appliquée avant <code>field_rename</code> . Dans le fichier INI, l'ordre de ces options est sans importance. Pour plus de clarté, spécifiez d'abord l'option <code>field_decoder</code> .
<code>next_parser</code>	<p>Le nom du prochain analyseur à exécuter. Cette option permet d'exécuter plusieurs analyseurs de façon séquentielle sur la même entrée.</p> <p><b>Note</b> Les analyseurs traitent tous les analyseurs suivants définis par le mot-clé <code>next_parser</code> et peuvent remplacer une valeur de champ déjà extraite par un analyseur précédent.</p>

Option	Description
exclude_fields	Une liste de noms de champs séparés par des points-virgules à supprimer de l'événement avant que ce dernier ne soit livré au serveur. Les noms de champs sont supprimés avant le filtrage des événements, afin que le champ que vous excluez lors de l'analyse ne puisse pas être utilisé dans la condition de filtre.
debug	Une option Oui ou Non qui permet de déboguer un analyseur donné. Lorsque le débogage est activé, l'analyseur effectue une journalisation détaillée de l'entrée qu'il a reçue, de l'opération exécutée et du résultat produit. Cette option s'applique par section, à savoir uniquement à l'analyseur défini par la section donnée. La valeur par défaut du débogage est debug=no pour les analyseurs.

### Configurations conditionnelles de l'option `field_decoder`

Pour les journaux ayant le même format commun, mais des différences importantes liées à des valeurs de champ spécifiques, des journaux avec les gravités **info** et **erreur** par exemple, vous pouvez utiliser l'analyseur imbriqué conditionnel pour réduire l'application d'analyseurs inutiles aux champs correspondants de journaux déjà analysés.

Par exemple, à l'aide de ces journaux :

```
2019-03-29T11:00:54.858Z host-FQDN Hostd: error hostd[2099230] [Originator@6876 sub=Default
opID=1983bdbe-cl-800f user=admin.user] AdapterServer caught exception: SSLExceptionE(SSL
Exception: error:140000DB:SSL routines:SSL routines:short read: The connection was closed by
the remote end during handshake.)
```

```
2019-03-29T11:00:55.477Z host-FQDN Hostd: info hostd[6D620B70] ['commonhost' opID=5759adcc-
cf] [transportConnector] -- FINISH task-internal-5726666 -- -- Completed connection restart
--
```

Vous pouvez utiliser la configuration suivante pour les analyser :

```
[parser|clf_parser]
base_parser=clf
format=%t %{generator_host}i %i: %{log_severity}i %i[%{thread_id}]i%M
field_decoder={"log_message" : {"log_severity" : {"error" : "error_parser", "info" :
"info_parser"}}}
exclude_fields=log_message

[parser|info_parser]
base_parser=clf
format=[%{common_info}i] [%{process}i] %M
field_rename={"log_message" : "info_log_content"}

[parser|error_parser]
base_parser=clfformat=[%{common_info}i] %{exception_handler}i %i:%{exception_type}i:i:%
```

```
{error_id}i:%i:%i:%i: %M
field_rename={"log_message" : "exception_content"}
```

Cette configuration produit les résultats suivants :

```
timestamp=2019-03-29T11:00:54.858000 generator_host="host-FQDN" log_severity="error"
thread_id="2099230" common_info=Originator@6876 sub=Default opID=1983bdbbe-c1-800f
user=admin.user exception_handler="AdapterServer" exception_type="SSLException(SSL
Exception" error_id="140000DB" exception_content="The connection was closed by the remote end
during handshake.)"
```

De plus, les champs suivants sont analysés pour le journal **info** :

```
timestamp=2019-03-29T11:00:55.477000 generator_host="host-FQDN" log_severity="info"
thread_id="6D620B70" log_message="['commonhost' opID=5759adcc-cf] [transportConnector]
-- FINISH task-internal-5726666 -- -- Completed connection restart --"
common_info="'commonhost' opID=5759adcc-cf" process="transportConnector" info_log_content="--
FINISH task-internal-5726666 -- -- Completed connection restart --"
```

## Analyseurs de journaux à valeurs séparées par des virgules (CSV, Comma-Separated Value)

Vous pouvez configurer des analyseurs Comma-Separated Value (CSV) pour les collecteurs FileLog et WinLog.

Les options disponibles pour l'analyseur `csv` sont `fields` et `delimiter`.

### Options de l'analyseur CSV

Les informations relatives à la structure de l'analyseur `csv` sont les suivantes :

Option	Description
<code>fields</code>	<p>L'option <code>fields</code> spécifie les noms des champs présents dans le journal. Le nombre total de noms de champs listés doit correspondre au nombre total de champs séparés par une virgule dans le journal. L'option <code>fields</code> est obligatoire pour l'analyseur CSV. Si vous ne la spécifiez pas, aucun objet n'est analysé. Les guillemets qui encadrent la valeur du champ sont facultatifs et dépendent du contenu du champ.</p> <p>Les noms de champs doivent être séparés par une virgule, comme suit :</p> <pre>fields = field_name1, field_name2, field_name3, field_name4</pre> <p>Cette définition suppose que les noms <code>field_name1</code>, <code>field_name2</code>, <code>field_name3</code> et <code>field_name4</code> sont attribués de manière séquentielle aux champs extraits.</p> <p>Si des champs sont omis par l'analyseur CSV, leurs noms peuvent également ne pas figurer sur la liste. Par exemple,</p> <pre>fields = field_name1, , field_name3, field_name4</pre> <p>Dans ce cas, l'analyseur extrait uniquement le premier, le troisième et le quatrième champs de l'événement. Il leur attribue ensuite les noms <code>field_name1</code>, <code>field_name3</code> et <code>field_name4</code>.</p> <p>Si l'option Champs ne spécifie pas une liste complète de champs dans vos journaux, l'analyseur renvoie une liste vide. Par exemple, si le fichier journal contient <code>field1</code>, <code>field2</code>, <code>field3</code>, <code>field4</code> et <code>field5</code>, mais que seul <code>fields= field1,field2,field3</code> est spécifié, l'analyseur renvoie une liste de champs vide.</p> <p>Vous ne pouvez pas utiliser <code>fields=*</code> pour un analyseur CSV, car l'analyseur renvoie une liste de champs vide. Une liste de champs complète doit être spécifiée, sauf si vous souhaitez omettre certains champs, comme décrit précédemment.</p>
<code>delimiter</code>	<p>L'option <code>delimiter</code> spécifie le délimiteur que l'analyseur doit utiliser. Par défaut, l'analyseur <code>csv</code> utilise une virgule comme délimiteur ; cependant, vous pouvez utiliser un point-virgule, un espace ou un autre caractère spécial comme délimiteur. Le délimiteur défini doit être placé entre guillemets. Par exemple, <code>delimiter=","</code> et <code>delimiter=";"</code>.</p> <p>L'analyseur <code>csv</code> prend en charge n'importe quel ensemble de caractères comme délimiteurs placés entre guillemets, par exemple <code>" "</code> ou <code>"asd"</code>. Les séparateurs de valeurs des champs dans les journaux doivent correspondre exactement au modèle défini par le paramètre des délimiteur, sinon l'analyseur va échouer.</p> <p>Des caractères spéciaux tels qu'un espace ou une tabulation peuvent être définis comme délimiteur pour l'analyseur <code>csv</code> à condition que le caractère d'échappement précède le caractère spécial (<code>\</code>, <code>\s</code>, <code>\t</code>). Par exemple, <code>delimiter="\s"</code> et <code>delimiter=" "</code>.</p> <p>L'option <code>delimiter</code> est facultative.</p>

## Configuration de l'analyseur de journaux CSV

Pour analyser des journaux issus de sources `winlog` ou `filelog`, utilisez la configuration suivante :

```
[filelog|some_csv_logs]
directory=D:\Logs
include=*.txt;*.txt.*
parser=myparser

[parser|myparser]
base_parser = csv
```

```
fields = timestamp,field_name1, field_name2, field_name3
delimiter = ";"
field_decoder={"timestamp": "tsp_parser"}
[parser|tsp_parser]
; timestamp is a built-in parser
base_parser=timestamp
; "format" is an option of timestamp parser
format=%Y-%m-%d %H:%M:%S
```

Selon cette configuration, les journaux collectés à partir de la source `some_csv_logs` (par exemple à partir du répertoire `directory=D:\Logs`) sont analysés par `myparser`. Si ces journaux contiennent trois valeurs séparées par un point-virgule, les événements analysés de façon séquentielle reçoivent les noms `field_name1`, `field_name2` et `field_name3`.

Pour analyser le journal CSV suivant :

```
"United States","USA","North America","High income: OECD","Fiscal year end: September 30;
reporting period for national accounts data: CY."
```

Définissez la configuration de l'analyseur CSV :

```
[parser|csv_log_parser]
base_parser=csv
fields=country_name, country_code, region, income_group, special_notes
```

L'analyseur CSV renvoie les champs suivants :

```
country_name=United States
country_code=USA
region=North America
income_group=High income: OECD
special_notes=Fiscal year end: September 30; reporting period for national accounts data: CY.
```

## Analyseur de journaux Common Log Format (Apache)

Vous pouvez configurer l'analyseur Apache Common Log Format (CLF) pour les collecteurs `FileLog` et `WinLog`.

### Analyseur Common Log Format (Apache)

L'analyseur Common Log Format (CLF) par défaut définit l'ordre et les noms de champs suivants :

```
host ident authuser datetime request statuscode bytes
```

Nom de l'analyseur : `clf`

L'option spécifique à l'analyseur CLF est `format`.

### Option Format

L'option `format` spécifie le format auquel les journaux Apache sont générés. Cette option n'est pas obligatoire.

Si vous ne spécifiez aucun format, le format commun par défaut suivant est utilisé.

```
%h %l %u %t \"%r\" %s %b
```

La chaîne de format de l'analyseur CLF n'accepte pas les expressions regex. Par exemple, spécifiez un espace au lieu de l'expression `\s+`.

Pour analyser d'autres formats de journaux, spécifiez-les dans la configuration de l'agent. Les champs analysés s'affichent côté serveur sous les noms suivants :

**Note** Si une variable est requise et que la variable `{VARIABLE}` est omise de la configuration, les champs sont ignorés.

Champs	Valeur
'%a':	« remote_ip »
'%A':	« local_ip »
'%B', '%b':	« response_size »
'%C':	dépend du nom de variable spécifié dans le format
'%c':	dépend du nom de variable spécifié dans le format
'%D':	« request_time_mcs »
'%E':	« error_status »
'%e':	dépend du nom de variable spécifié dans le format
'%F', '%f':	« file_name »
'%h':	« remote_host »
'%H':	« request_protocol »
'%i':	dépend du nom de variable spécifié dans le format
'%k':	« keepalive_request_count »
'%l':	« remote_log_name »
'%L'	« request_log_id »
'%M':	« log_message » (l'analyseur cesse d'analyser le journal d'entrée lorsque ce spécificateur est atteint)
'%m':	« request_method »
'%n':	dépend du nom de variable spécifié dans le format
'%o':	dépend du nom de variable spécifié dans le format

Champs	Valeur
'%p':	« server_port » Des formats supplémentaires peuvent être utilisés avec ce spécificateur : % <i>{format}</i> p. Les formats pris en charge sont « canonical », « local » ou « remote ». Lorsque le format « canonical » est utilisé, le nom de champ reste « server_port ». Lorsque le format « local » est utilisé, le nom de champ est « local_server_port » et, lorsque le format « remote » est utilisé, le nom de champ est « remote_server_port ».
'%P':	« process_id » Des formats supplémentaires peuvent être utilisés avec ce spécificateur : % <i>{format}</i> P. Les formats pris en charge sont « pid », « tid » et « hextid ». Si « pid » est utilisé comme format, le nom de champ est « process_id ». Les formats « tid » et « hextid » génèrent des champs avec le nom « thread_id »
'%q':	« query_string »
'%r':	« request »
'%R':	« response_handler »
'%s':	« status_code ». qui génère l'état final de la demande, est également pris en charge. Il apparaît sur le serveur sous la forme « status_code ».

Champs	Valeur
'%t':	<p>« timestamp » fonctionne en tant qu'horodatage d'événement lors de l'ingestion, engage l'analyseur horodatage. Pour remplacer la détection automatique d'horodatage, le format de date et d'heure peut être spécifié entre accolades : %Y-%m-%d %H:%M:%S)t, reportez-vous à la section <a href="#">Analyseur Timestamp</a> pour plus de détails.</p> <p>Le format d'horodatage de l'analyseur CLF peut commencer par le préfixe "begin:" ou "end:". Si le format commence par begin: (par défaut), l'heure est prise au début du traitement de la demande. S'il commence par end:, l'heure est prise au moment de l'écriture de l'entrée de journal, vers la fin du traitement de la demande. Par exemple, les formats tels que les suivants sont pris en charge par l'analyseur CLF : %h %l %u [% {begin:%d/%b/%Y %T}t. %{msec_frac}t] \"%r\" %&gt;s %b</p> <p>Les jetons de format suivants sont également pris en charge pour le spécificateur du format d'horodatage de l'analyseur CLF :</p> <p><b>sec</b></p> <p>nombre de secondes depuis Epoch. Équivalent au spécificateur %s de l'analyseur Horodatage.</p> <p><b>msec</b></p> <p>nombre de millisecondes depuis Epoch</p> <p><b>usec</b></p> <p>nombre de microsecondes depuis Epoch</p> <p><b>msec_frac</b></p> <p>fraction de milliseconde (équivalent au spécificateur %f de l'analyseur Horodatage)</p> <p><b>musec</b></p> <p>fraction de microseconde (équivalent au spécificateur %f de l'analyseur Horodatage)</p> <p>Pour analyser des journaux dans lesquels l'horodatage est représenté avec des jetons de format, les formats suivants peuvent être utilisés dans la configuration :</p> <pre>format=%h %l %u %{sec}t \"%r\" %s %b format=%h %l %u %{msec}t \"%r\" %s %b format=%h %l %u %{usec}t \"%r\" %s %b</pre> <p>Ces jetons ne peuvent pas être combinés entre eux ou avec le formatage de l'analyseur Horodatage dans la même chaîne de format. Vous pouvez utiliser plusieurs jetons % {format}t à la place. Par exemple, pour utiliser Horodatage qui inclut des millisecondes, à l'exception de l'utilisation du spécificateur %f de l'analyseur Horodatage, l'horodatage combiné suivant peut être utilisé : %d/%b/%Y %T}. %{msec_frac}t .</p>
'%T':	« request_time_sec »
'%u':	« remote_auth_user »
'%U':	« requested_url »
'%v':	« server_name »
'%V':	« self_referential_server_name »
'%X':	« connection_status » dépend du nom de variable spécifié dans le format

Champs	Valeur
'%x':	dépend du nom de variable spécifié dans le format
'%I':	« received_bytes »
'%O':	« sent_bytes »
'%S':	« transferred_size »

Par exemple, pour analyser les journaux collectés à partir des sources `winlog` ou `filelog` à l'aide de l'analyseur CLF, spécifiez la configuration suivante :

```
[filelog|clfflogs]
directory=D:\Logs
include=*.txt
parser=myclf

[parser|myclf]
debug=yes ;Note: use this option only while debugging and set it to 'no' when used in
production.
base_parser=clf
format=%h %l %u %b %t \"%r\" %s
```

Selon cette configuration, les journaux collectés à partir de la source `clfflogs`, par exemple à partir du répertoire `directory=D:\Logs`, sont analysés par `myclf`. L'analyseur `myclf` traite uniquement les journaux générés au format décrit dans la configuration.

La valeur par défaut du débogage est `debug=no` pour les analyseurs.

### Analyse des journaux générés à l'aide du format CLF

Pour analyser les journaux générés à l'aide du format CLF, définissez le format correspondant dans la configuration. Par exemple,

```
format=%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User_Agent}i\"
```

Les champs remplis qui utilisent les spécificateurs `%{Referer}i` et `%{User_Agent}i` s'affichent sur le serveur vRealize Log Insight sous les noms `referer` et `user_agent`, respectivement.

### Intégration de l'analyseur timestamp à l'analyseur CLF

Vous pouvez analyser des journaux Apache à un format d'heure personnalisée.

Accédez aux journaux dont le format d'heure personnalisée est le suivant :

```
format = %h %l %u %{%a, %d %b %Y %H:%M:%S}t \"%r\" %>s %b
```

Si aucune heure personnalisée n'est spécifiée, l'analyseur CLF tente de déduire automatiquement le format d'heure en exécutant l'analyseur timestamp automatique. Dans le cas contraire, le format d'heure personnalisée est utilisé.

Les formats d'heures personnalisées pris en charge pour les journaux d'erreurs sont les suivants :

Format d'heure personnalisée	Description	Format de configuration
%{u}t	Heure actuelle microsecondes incluses	format=[%{u}t] [%l] [pid %P] [client %a] %M
%{cu}t	Heure actuelle au format compact ISO 8601 microsecondes incluses	format=[%{cu}t] [%l] [pid %P] [client %a] %M

Pour obtenir la liste complète des spécificateurs d'horodatage pris en charge, consultez la rubrique [Analyseur Timestamp](#).

### Exemple : Configuration des journaux d'accès Apache par défaut pour Windows

### Exemple : Configuration des journaux d'erreurs Apache par défaut pour Windows

Cet exemple montre comment formater des configurations de journaux d'accès Apache v2.4 pour Windows.

```
;ACCESS LOG
;127.0.0.1 - - [13/May/2015:14:44:05 +0400] "GET /xampp/navi.php HTTP/1.1" 200 4023 "http://localhost/xampp/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:37.0) Gecko/20100101 Firefox/37.0"
;format=%h %l %u %{d/%b/%Y:%H:%M:%S %z}t \"%r\" %>s %b \"%{Referer}i\" \"%{User_agent}i\"

; Section to collect Apache ACCESS logs
[filelog|clfflogs-access]
    directory=C:\xampp\apache\logs
    include=acc*
    parser=clfparser_apache_access
    enabled=yes

;Parser to parse Apache ACCESS logs
[parser|clfparser_apache_access]
    debug=yes
    base_parser=clf
    format=%h %l %u %{d/%b/%Y:%H:%M:%S %z}t \"%r\" %>s %b \"%{Referer}i\" \"%{User_agent}i\"
```

Définissez le format des journaux d'accès :

#### 1 Configurez Apache pour le format des journaux d'accès (httpd.conf) :

```
LogFormat "%h %l %u %{d-%b-%Y:%H:%M:%S}t \"%r\" %a %A %e %k %l %L %m %n %T %v %V %>s %b \"%{Referer}i\" \"%{User_Agent}i\"" combined
```

#### 2 Définissez la configuration de l'analyseur CLF :

```
;ACCESS LOG
;127.0.0.1 unknown - 21-May-2015:13:59:35 "GET /xampp/navi.php HTTP/1.1" 127.0.0.1 127.0.0.1
- 0 unknown - GET - 1 localhost localhost 200 4023 "http://localhost/xampp/" "-"
[filelog|clfflogs-access]
    directory=C:\xampp\apache\logs
    include=acc*,_myAcc*
    parser=clfparser_apache_access
    enabled=yes

; Parser to parse Apache ACCESS logs
[parser|clfparser_apache_access]
```

```

debug=yes
base_parser=clf
format=%h %l %u %{d-%b-%Y:%H:%M:%S}t \"%r\" %a %A %e %k %l %L %m %n %T %v %V %>s %b \"%
{Referer}i\" \"%{User_Agent}i\"

```

L'analyseur CLF renvoie les informations suivantes :

```

remote_host=127.0.0.1
timestamp=2015-05-13T10:44:05
request=GET /xampp/navi.php HTTP/1.1
status_code=200
response_size=4023
referer=http://localhost/xampp/
user_agent=Mozilla/5.0 (Windows NT 6.1; WOW64; rv:37.0) Gecko/20100101 Firefox/37.0

```

Cet exemple montre comment formater des configurations de journaux d'erreurs Apache v2.4 pour Windows.

```

;ERROR LOG
;[Wed May 13 14:37:17.042371 2015] [mpm_winnt:notice] [pid 4488:tid 272] AH00354: Child:
Starting 150 worker threads.
;[Wed May 13 14:37:27.042371 2015] [mpm_winnt:notice] [pid 5288] AH00418: Parent: Created
child process 3480
;format=[%{a %b %d %H:%M:%S%f %Y}t] [%m:%{severity}i] [pid %P:tid %{thread_id}i] %E: %M
;format=[%{a %b %d %H:%M:%S%f %Y}t] [%m:%{severity}i] [pid %P] %E: %M

; Section to collect Apache ERROR logs
[filelog|clfflogs-error]
    directory=C:\xampp\apache\logs
    include=err*
    parser=clfparsed_apache_error
    enabled=yes

;Parser to parse Apache ERROR logs
[parser|clfparsed_apache_error]
    debug=yes
    base_parser=clf
    format=[%{a %b %d %H:%M:%S%f %Y}t] [%m:%{severity}i] [pid %P:tid %{thread_id}i] %E: %M
    next_parser=clfparsed_apache_error2

;Parser to parse Apache ERROR logs
[parser|clfparsed_apache_error2]
    debug=yes

```

```
base_parser=clf  
format=[%{a %b %d %H:%M:%S%f %Y)t] [%m:%{severity}i] [pid %P] %E: %M
```

**Note** Les noms fournis correspondent au format de journal combiné. Les journaux d'erreurs d'Apache sont également décrits à l'aide des clés de formatage précédentes, et non avec le format du journal d'erreurs d'Apache.

Définissez le format des journaux d'erreurs :

1 Configurez Apache pour le format des journaux d'erreurs (httpd.conf) :

```
LogFormat "%h %l %u %{d-%b-%Y:%H:%M:%S}t \"%r\" %a %A %e %k %l %L %m %n %T %v %V %>s %b\n\"%{Referer}i\" \"%{User_Agent}i\" combined
```

2 Définissez la configuration de l'analyseur CLF :

```
;Parser to parse Apache ERROR logs
[parser|clfparsers_apache_error]
  debug=yes
  base_parser=clf
  format=[%{a} %b %d %H:%M:%S%f %Y}t] [%m:%{severity}i] [pid %P] %E: %M
  next_parser=clfparsers_apache_error2

;Parser to parse Apache ERROR logs
[parser|clfparsers_apache_error2]
  debug=yes
  base_parser=clf
  format=[%{a} %b %d %H:%M:%S%f %Y}t] [%m:%{severity}i] [pid %P:tid %{thread_id}i] %E: %M
```

Entrée journal :

```
[Wed May 13 14:37:17.042371 2015] [mpm_winnt:notice] [pid 4488:tid 272] AH00354: Child: Starting 150 worker threads.
```

L'analyseur CLF renvoie les champs suivants pour l'entrée journal (si un analyseur est utilisé dans un fuseau horaire +0400) :

```
timestamp=2015-05-13T10:37:17.042371
request_method=mpm_winnt
severity=notice
process_id=4488
thread_id=272
error_status=AH00354
log_message=Child: Starting 150 worker threads.
```

Entrée journal :

```
[Wed May 13 14:37:27.042371 2015] [mpm_winnt:notice] [pid 5288] AH00418: Parent: Created child process 3480
```

L'analyseur CLF renvoie les champs suivants pour l'entrée journal (si un analyseur est utilisé dans un fuseau horaire +0400) :

```
timestamp=2015-05-13T10:37:27.042371
request_method=mpm_winnt
severity=notice
process_id=5288
error_status=AH00418
log_message=Parent: Created child process 3480
```

## Analyseur de paire clé-valeur

Vous pouvez configurer l'analyseur de paire clé-valeur (KVP) pour les collecteurs `FileLog` et `WinLog`.

### Analyseur de paire clé-valeur

L'analyseur `kvp` recherche toutes les correspondances `key=value` dans un texte de message journal arbitraire et les extrait. L'exemple suivant illustre le format de l'analyseur `kvp`.

```
[parser|kvp_parser]
base_parser=kvp
fields=*
```

Par exemple, le journal clé-valeur peut être au format suivant : `scope=local; abstract=false; lazyInit=false; autowireMode=0; dependencyCheck=0;`

Avec l'analyseur `kvp`, vous devez spécifier les champs dont les valeurs doivent être extraites. Par exemple, si la définition `fields=name,lastname,country` existe dans la configuration, seules les valeurs comportant les clés spécifiées sont analysées et envoyées vers le serveur.

La clé et la valeur peuvent éventuellement être placées entre guillemets (" ") pour définir un espace ou d'autres caractères spéciaux.

Lorsque des guillemets doubles sont utilisés pour la clé ou la valeur, le caractère barre oblique inverse « \ » peut être utilisé comme caractère d'échappement. Tout caractère suivant le caractère barre oblique inverse est défini littéralement, notamment le caractère guillemet double ou le caractère barre oblique inverse. Par exemple : « \\ »

Notez les points suivants :

- Si la paire clé-valeur n'est pas suivie d'un signe égal et que l'extrait de code `VALUE` n'est pas fourni, l'option est ignorée et traitée comme un texte libre.
- La clé ne peut pas être vide, mais la valeur peut l'être.
- Si un signe égal n'est suivi d'aucune valeur, il est ignoré et traité comme un texte libre.
- Une valeur peut être une chaîne de caractères encadrés de guillemets ou elle peut être vide. Pour l'échappement de caractères spéciaux faisant partie de la valeur, utilisez une barre oblique inverse.

### Options de l'analyseur KVP

Les informations relatives à la structure de l'analyseur `kvp` sont les suivantes :

Option	Description
<code>fields</code>	<p>Informations à extraire décrites comme unités de données. Par exemple, <code>fields=name,lastname,country</code>.</p> <p>Si des noms de champs spécifiques sont définis par l'option <code>fields</code>, chaque caractère non valide d'un nom de champ extrait d'un journal est remplacé par un trait de soulignement. Par exemple, si l'option <code>fields</code> cherche les champs « x-A » et « a*(X+Y) », l'analyseur extrait ces champs des journaux et les renomme respectivement « x_a » et « a__x_y ». Cela permet d'extraire des champs dont le nom contient tout type de caractère.</p> <p>Si l'option <code>fields</code> est définie sur « * », l'analyseur <code>kvp</code> reconnaît automatiquement les paires champ/valeur et il recherche les champs comportant uniquement des caractères « alphanumériques + trait de soulignement » (pris en charge par le serveur LI). Tous les autres caractères non valides sont rejetés au lieu d'être convertis en traits de soulignement. Cela empêche l'analyseur d'extraire des informations inutiles dans les champs statiques.</p>
<code>delimiter</code>	<p>Cette option est facultative.</p> <p>Les délimiteurs par défaut sont les caractères espace, tabulation, nouvelle ligne, virgule et point-virgule.</p> <p>Si aucun délimiteur n'est spécifié dans la configuration, l'analyseur <code>kvp</code> utilise les délimiteurs par défaut.</p> <p>Pour modifier les délimiteurs par défaut, vous devez définir des délimiteurs spécifiques entre guillemets doubles. Par exemple : <code>delimiter = "#^ "</code>. Selon cette définition, chaque caractère encadré de guillemets sert de délimiteur. Pour l'analyseur <code>kvp</code>, tout caractère peut être utilisé comme délimiteur. Vous pouvez inclure les délimiteurs par défaut et d'autres délimiteurs dans la définition.</p> <p>Par exemple, l'instruction <code>delimiter = "#^ \t\r\n\s"</code> inclut les caractères tabulation, nouvelle ligne, ainsi que l'espace comme délimiteurs. Si ces caractères sont utilisés, ils doivent être précédés du caractère d'échappement. Par exemple, pour définir le caractère espace comme délimiteur, entrez le caractère d'échappement « \ » avant le caractère espace lors de sa définition comme délimiteur, par exemple, <code>delimiter="\s"</code>.</p>
<code>field_decoder</code>	<p>Les analyseurs imbriqués spécifiés en tant que chaîne JSON, dans laquelle les clés sont les noms des champs auxquels appliquer l'analyseur imbriqué et la valeur est le nom de l'analyseur à utiliser pour ce champ.</p> <p>Chaque analyseur imbriqué est appliqué au champ approprié, tel qu'il est décodé par l'analyseur de base.</p> <p>Les décodeurs de champs sont utiles lorsque la valeur d'une paire clé-valeur est complexe, telle qu'un horodatage ou une liste séparée par des virgules.</p>
<code>debug =</code>	<p>Cette option est facultative. La valeur <code>debug =</code> peut être <code>yes</code> ou <code>no</code>. La valeur par défaut du débogage est <code>debug=no</code> pour les analyseurs.</p> <p>Lorsque cette option est définie sur <code>yes</code>, vous pouvez afficher les journaux détaillés de l'ingestion de l'analyseur dans <code>liagent_&lt;date&gt;.log</code>.</p>

## Options de valeurs de clés supplémentaires

Clé	Définition
<code>KVP_MESSAGE = *(MESSAGE_ENTRY [WSPR])</code>	Liste d'entrées de messages séparées par un espace facultatif
<code>MESSAGE_ENTRY = KVP / FREE_TEXT</code>	Une entrée est une paire clé-valeur ou simplement du texte libre

Clé	Définition
KVP = KEY ["=" VALUE]	Paire clé-valeur. Si KEY n'est pas suivi d'un signe égal et de VALUE, il est ignoré comme un texte libre.
KEY = BARE_KEY / QUOTED_KEY	
FREE_TEXT = "="	Un signe égal isolé est considéré comme du texte libre et ignoré.
BARE_KEY = *1BARE_KEY_CHAR	Au moins un caractère
BARE_KEY_CHAR = %0x00-08 / %0x10-19 / %0x21-3C / %3E-%FF	Tout caractère à l'exclusion du signe égal, du caractère espace ou du caractère tabulation
QUOTED_KEY = 0x22 *1(QUOTED_STRING_CHAR / "\" CHAR) 0x22	Au moins un caractère placé entre guillemets doubles. La barre oblique inverse est utilisée comme caractère d'échappement.
QUOTED_STRING_CHAR = %0x00-21 / %0x23-FF	Tout caractère à l'exclusion du guillemet double
VALUE = BARE_VALUE / QUOTED_VALUE	
BARE_VALUE = *BARE_VALUE_CHAR	Aucun caractère ou plus
BARE_VALUE_CHAR = %0x00-08 / %0x10-19 / %0x21-FF	Tout caractère à l'exclusion du caractère espace ou du caractère tabulation
QUOTED_VALUE = 0x22 *(QUOTED_STRING_CHAR / "\" CHAR) 0x22	Chaîne de caractères placés entre guillemets doubles. Elle peut être vide. La barre oblique inverse est utilisée comme caractère d'échappement.

## Exemples de configurations d'analyseur KVP

Si nécessaire, vous pouvez utiliser `fields=*` pour analyser tous les champs.

```
[parser|simple_kv]
base_parser =kvp
fields=*
```

Cet exemple montre comment spécifier le décodeur de champs.

```
[parser|mykvp]
debug=no
base_parser=kvp
delimiter="#"^|"
fields=*
;OR fields=scope,abstract,lazyInit,autowireMode,dependencyCheck
field_decoder={"field1":"field1_parser1"}

[parser|field1_parser1]
base_parser=clf
format=[%{value1}i]]
field_decoder={"value1":"field_parser2"}
```

Pour analyser le journal KVP suivant :

```
Configuring transport... proto = cfapi server_hostname = LOCALHOST ssl = no port = 9000
reconnect = 30
```

Définissez la configuration de l'analyseur KVP :

```
[parser|kvp_log_parser]
base_parser=kvp
fields=*
```

L'analyseur KVP renvoie les champs suivants :

```
proto=cfapi
server_hostname=LOCALHOST
ssl=no
port=9000
reconnect=30
```

### Exemple : Exemples d'analyseurs KVP simple et complexe

Exemple d'analyseur KVP simple

```
[filelog|MyLog]
directory=C:\<folder_name>\Parser_logs
include=*.log
parser=my_KVP_parser

[parser|my_KVP_parser]
base_parser=kvp
fields=*
```

Exemple d'analyseur KVP complexe

```
[filelog|MyLog]
directory=C:\<folder_name>\Parser_logs
include=*.log
parser=my_KVP_parser

[parser|my_KVP_parser]
base_parser=kvp
fields=*
field_decoder={"field1":" field1_parser1"}

[parser| field1_parser1]
base_parser=clf
format=[%{value1}i]]
field_decoder={"value1":" field1_parser2"}
```

## Analyseur Timestamp

L'analyseur `timestamp` ne produit pas de champs, mais transforme les entrées du format de chaîne au format d'horodatage interne affiché en millisecondes à compter du début de l'époque UNIX, le 1er janvier 1970 (minuit UTC/GMT).

L'unique option de configuration prise en charge est `format`. Par exemple, `format=%Y-%m-%d %H:%M:%S`.

Contrairement à l'analyseur Common Log Format (CLF), l'analyseur `timestamp` peut analyser l'heure lorsque les spécificateurs d'heures ne sont séparés par aucun délimiteur, comme suit :  
`%A%B%d%H%M%S%Y%z`.

Les spécificateurs de formats utilisés par l'analyseur `timestamp` sont les suivants :

```
'%a': Abbreviated weekday name, for example: Thu
'%A': Full weekday name, for example: Thursday
'%b': Abbreviated month name, for example: Aug
'%B': Full month name, for example: August
'%d': Day of the month, for example: 23. strftime() expects zero-padded (01-31) digits
for this specifier but Log Insight agents can parse space-padded and non-padded
day numbers, too.
'%e': Day of the month, for example: 13. strftime() expects space-padded ( 1-31) digits
for this specifier but Log Insight agents can parse zero-padded and non-padded
day numbers too.
'%f': Fractional seconds of time, for example: .036 'f' specifier assumes that '.' or ','
character should exist before fractional seconds and there is no need to mention
that character in the format. If none of these characters precedes fractional
seconds,
timestamp wouldn't be parsed.
'%H': Hour in 24h format (00-23), for example: 14. Zero-padded, space-padded, and non-
padded hours
are supported.
'%I': Hour in 12h format (01-12), for example: 02. Zero-padded, space-padded and non-
padded hours
are supported.
'%m': Month as a decimal number (01-12), for example: 08. Zero-padded, space-padded
and non-padded month numbers are supported.
'%M': Minute (00-59), for example: 55
'%p': AM or PM designation, for example: PM
'%S': Second (00-61), for example: 02
'%s': Total number of seconds from the UNIX epoch start, for example 1457940799
(represents '2016-03-14T07:33:19' timestamp)
'%Y': Year, for example: 2001
'%z': ISO 8601 offset from UTC in timezone (1 minute=1, 1 hour=100).., for example: +100
```

Des spécificateurs supplémentaires sont acceptés par l'analyseur Timestamp, mais leurs valeurs sont ignorées et n'affectent pas l'heure analysée.

```
'%C': Year divided by 100 and truncated to integer (00-99), for example: 20
'%g': Week-based year, last two digits (00-99), for example, 01
'%G': Week-based year, for example, 2001
'%j': Day of the year (001-366), for example: 235
```

```
'%u':    ISO 8601 weekday as number with Monday as 1 (1-7), for example: 4
'%U':    Week number with the first Sunday as the first day of week one (00-53), for example:
33
'%V':    ISO 8601 week number (00-53), for example: 34
'%w':    Weekday as a decimal number with Sunday as 0 (0-6), for example: 4
'%W':    Week number with the first Monday as the first day of week one (00-53), for example:
34
'%y':    Year, last two digits (00-99), for example: 01
```

Si un paramètre `format` n'est pas défini, l'analyseur `Timestamp` analyse les horodatages selon les formats par défaut.

### Analyseur Automatic Timestamp

L'analyseur Automatic Timestamp est appelé lorsqu'aucun format n'est défini pour l'analyseur Timestamp. L'analyseur peut aussi être appelé directement sans définition d'analyseur Timestamp en utilisant `timestamp` dans `field_decoder`. Par exemple :

```
[parser|mycsv]
base_parser=csv
debug=yes
fields=timestamp,action,source_id,dest
field_decoder={"timestamp": "timestamp"}
```

### Exemple : Analyseur timestamp avec la configuration par défaut

Cet exemple montre un analyseur `timestamp` avec une configuration par défaut.

```
[parser|tsp_parser]
base_parser=timestamp
debug=no
format=%Y-%m-%d %H:%M:%S%f
```

Pour intégrer un analyseur `timestamp` à d'autres analyseurs, tels que les analyseurs Comma-Separated Value (CSV), spécifiez la configuration suivante :

```
[parser|mycsv]
base_parser=csv
fields=timestamp,action,source_id,dest
field_decoder={"timestamp": "tsp_parser"}
```

Lorsque cette configuration est définie, l'analyseur `mycsv` extrait les champs qui incluent les noms spécifiés dans la configuration et exécute `tsp_parser` sur le contenu du champ `timestamp`. Si `tsp_parser` récupère un horodatage valide, le serveur utilise cet horodatage pour le message de journal.

### Analyseur de journal automatique

L'analyseur détecte automatiquement l'horodatage dans les 200 premiers caractères d'une ligne. Le format des horodatages détectés automatiquement est identique à celui de l'analyseur `timestamp`.

L'analyseur automatique ne possède aucune option. En plus de la détection automatique de l'horodatage, l'analyseur clé-valeur s'exécute sur l'entrée de journal et détecte automatiquement les paires clé-valeur existantes dans les journaux. Il procède ensuite à l'extraction des champs en conséquence. Par exemple,

```
[filelog|some_logs]
directory=/var/log
include=*
parser=auto
```

À l'instar d'autres analyseurs, vous pouvez définir une action distincte pour l'analyseur automatique.

```
[filelog|kvplogs]
directory=C:\temp_logs\csv-itbm
include=*.txt
parser=myauto
[parser|myauto]

base_parser=auto
debug=yes
```

Si vous avez activé l'extrait de code `debug` pour l'analyseur automatique, des informations supplémentaires relatives à l'analyse sont imprimées. Ces informations concernent notamment l'identité du journal sur lequel l'analyseur automatique a été exécuté et celle des champs extraits du journal.

La valeur par défaut du débogage est `debug=no` pour les analyseurs.

## Analyseur Syslog

L'analyseur Syslog prend en charge les options `message_decoder` et `extract_sd` et détecte automatiquement trois formats : RFC-6587, RFC-5424 et RFC-3164.

### Configuration de l'option `message_decoder`

Toutes les options courantes, ainsi que l'option `message_decoder` sont disponibles pour l'analyseur Syslog. Par défaut, seuls les champs `timestamp` et `appname` sont extraits. Activez l'option `message_decoder` en définissant les valeurs de configuration du fichier `liagent.ini` comme indiqué dans l'exemple suivant :

```
[filelog|data_logs]
directory=D:\Logs
include=*.txt
parser=mysyslog

[parser|mssyslog]
base_parser=syslog
message_decoder=syslog_message_decoder
debug=yes

[parser|syslog_message_decoder]
```

```
base_parser=kvp
fields=*
```

### Exemple : Analyse avec l'option message\_decoder

L'exemple suivant montre un exemple d'événement et les champs ajoutés à l'événement par un analyseur Syslog configuré pour utiliser l'option message\_decoder :

- Exemple d'événement :

```
2015-09-09 13:38:31.619407 +0400 smith01 john: Fri Dec 5 08:58:26 2014 [pid 26123]
[jsmith.net] status_code=FAIL oper_
ation=LOGIN: Client "176.31.17.46"
```

- Renvoyé par un analyseur Syslog auquel l'option message\_decoder est appliquée pour exécuter un analyseur KVP :

```
timestamp=2015-09-09T09:38:31.619407 appname=john status_code=FAIL operation=LOGIN:
```

### Configuration de l'option extract\_sd pour l'analyse de données structurées

Pour analyser des données structurées, activez l'option extract\_sd en définissant les valeurs de configuration du fichier liagent.ini comme indiqué dans l'exemple suivant :

```
[filelog|simple_logs]
directory=/var/log
include=*.txt
parser=syslog_parser

[parser|syslog_parser]
base_parser=syslog
extract_sd=yes
```

### Exemple : Analyse avec l'option extract\_sd

L'exemple suivant montre un exemple d'événement et les champs ajoutés à l'événement par un analyseur Syslog configuré pour utiliser l'option extract\_sd :

- L'exemple d'événement : <165>1 2017-01-24T09:17:15.719Z localhost evntslog - ID47 [exampleSDID@32473 iut="3" eventSource="Application" eventID="1011"] [examplePriority@32473 class="high"] Found entity IPSet, display name dummy ip set 1411

- Les champs suivants sont ajoutés à l'événement par l'analyseur Syslog :

```
timestamp=2017-01-24T09:17:15.719000
pri_facility=20
pri_severity=5
procid="-"
msgid="ID47"
iut="3"
```

```

eventsource="Application"
eventid="1011"
class="high"
appname="evntslog"

```

## Champs extraits par l'analyseur

L'analyseur extrait automatiquement les champs suivants à partir d'un événement :

Classification RFC	pri_facility	pri_severity	timestamp	appname	procid	msgid
Non-RFC			X	X		
RFC-3164	X	X	X	X		
RFC-5424	X	X	X	X	X	X

## Options de l'analyseur Syslog

Le tableau suivant décrit les options Syslog disponibles.

Option	Description
message_decoder	Définit un analyseur supplémentaire qui est utilisé pour analyser le corps de message d'un événement. Il peut s'agir d'un analyseur intégré, tel que « auto » ou de tout autre analyseur personnalisé.
extract_sd	Analyse les données structurées. Seules les valeurs « yes » ou « no » sont compatibles avec l'option extract_sd. L'option est désactivée par défaut. Lorsque l'option extract_sd est activée, elle extrait simplement toutes les paires clé-valeur des données structurées.

## Exemple : Analyse pour la norme RFC-5424

Les exemples suivants montrent deux événements analysés par une instance Syslog configurée, la configuration utilisée pour le collecteur, un exemple d'événement et les champs que l'analyseur Syslog ajoute à l'événement.

### ■ Configuration :

```

[filelog|simple_logs]
directory=/var/log
include=*.txt
parser=syslog

```

### ■ Un événement est généré dans le fichier surveillé :

```

<165>1 2017-01-24T09:17:15.719Z router1 mgd 3046 UI_DBASE_LOGOUT_EVENT
[junos@2636.1.1.1.2.18 username=\"regress\"] User 'regress' exiting configuration
mode - Juniper format

```

- Champs ajoutés à l'événement par l'analyseur Syslog :

```
The following fields will be added to the event by Syslog parser:
timestamp=2017-01-24T09:17:15.719000
pri_facility = 20
pri_severity = 5
procid = 3046
msgid = UI_DBASE_LOGOUT_EVENT
appname = mgd
```

### Exemple : Analyse pour la norme RFC-3164

L'exemple suivant montre la configuration utilisée pour le collecteur, un exemple d'événement RFC-3164 et les champs que Syslog ajoute à l'événement.

- Configuration :

```
[filelog|simple_logs]
directory=/var/log
include=*.txt
parser=syslog
```

- Un événement RFC-3164 généré dans le fichier surveillé :

```
<13>2017-01-24T09:17:15.719Z router1 mgd 3046 UI_DBASE_LOGOUT_EVENT User 'regress' exiting
configuration mode - Juniper format
```

- Champs ajoutés à l'événement par l'analyseur Syslog :

```
timestamp=2017-01-24T09:17:15.719000
pri_facility=1
pri_severity=5
appname="mgd"
```

## Analyseur LTSV

Le format LTSV (Labeled Tab-Separated Values) est une variante du format TSV (Tab-Separated Values).

Dans un fichier LTSV, chaque enregistrement est représenté par une ligne unique. Chaque champ est séparé par <TAB> et contient un libellé et une valeur. Le libellé et la valeur sont séparés par :. Avec le format LTSV, vous pouvez analyser chaque ligne en la fractionnant avec <TAB> (comme pour le format TSV) et étendre n'importe quel champ avec des libellés uniques dans l'ordre que vous voulez. Pour plus d'informations sur le format LTSV, consultez <http://ltsv.org/>.

## Exemple : Configuration de l'analyseur LTSV

### Exemple : Exemple de journal LTSV

L'analyseur LTSV ne requiert pas d'options de configuration spécifiques. Pour l'utiliser, spécifiez le nom de l'analyseur `ltsv` intégré dans la configuration.

```
[parser|myltsv]
base_parser=ltsv
```

Le fichier LTSV doit être une séquence d'octets correspondant à la production LTSV au format ABNF.

```
ltsv = *(record NL) [record]
record = [field *(TAB field)]
field = label ":" field-value
label = 1*lbyte
field-value = *fbyte

TAB = %x09
NL = [%x0D] %x0A
lbyte = %x30-39 / %x41-5A / %x61-7A / "_" / "." / "-" ; [0-9A-Za-z_.-]
fbyte = %x01-08 / %x0B / %x0C / %x0E-FF
```

```
host:127.0.0.1<TAB>ident:-<TAB>user:frank<TAB>time:[10/Oct/2000:13:55:36 -0700]<TAB>req:GET /
apache_pb.gif HTTP/1.0<TAB>status:200<TAB>size:2326<TAB>referer:http://www.example.com/
start.html<TAB>ua:Mozilla/4.08 [en] (Win98; I ;Nav)
```

Avec l'exemple de configuration LTSV, l'analyse du journal doit renvoyer les champs suivants :

```
host=127.0.0.1
ident=-
user=frank
time=[10/Oct/2000:13:55:36 -0700]
req=GET /apache_pb.gif HTTP/1.0
status=200
size=2326
referer=http://www.example.com/start.html
ua=Mozilla/4.08 [en] (Win98; I ;Nav)
```

## Configuration de débogage

Un débogage supplémentaire est également disponible pour l'analyseur LTSV. Le débogage LTSV est désactivé par défaut. Pour l'activer, entrez `debug=yes`.

```
[parser|myltsv]
base_parser=ltsv
debug=yes
```

Lorsque le débogage est activé, l'analyseur LTSV extrait les valeurs de tous les libellés valides du journal. Avec l'analyseur LTSV, les noms de libellés doivent comporter uniquement des caractères alphanumériques, le trait de soulignement (" \_"), le point (".") et le tiret ("-"). Si le journal contient au moins un nom de libellé non valide, son analyse échoue. Même si le nom du libellé est valide, l'agent contrôle le nom du champ. S'il existe des noms non valides, le nom du libellé doit être corrigé en nom de champ valide.

### Configuration de l'analyseur LTSV de la section `filelog`

Vous pouvez aussi configurer directement l'analyseur LTSV de la section `filelog`.

```
[filelog|simple_logs]
directory=/var/log
include=*
parser=ltsv
```

## Analyseur Regex

L'analyseur `regex` permet l'utilisation de certaines expressions régulières pour les données collectées.

Les agents vRealize Log Insight utilisent l'expression régulière de bibliothèque C++ Boost, dont la syntaxe est Perl. L'analyseur `regex` peut être défini en spécifiant un modèle d'expression régulière contenant des groupes de capture nommés. Par exemple : `(?<field_1>\d{4}) [-] (?<field_2>\d{4}) [-] (?<field_3>\d{4}) [-] (?<field_4>\d{4})`

Les noms spécifiés dans les groupes (par exemple : `field_1`, `field_2`, `field_3` et `field_4`) deviennent les noms des champs extraits correspondants. Conditions requises des noms :

- Les noms spécifiés dans le modèle d'expression régulière doivent être des noms de champs valides pour vRealize Log Insight.
- Les noms ne peuvent contenir que des caractères alphanumériques et le caractère de soulignement « \_ ».
- Le nom ne peut pas commencer par un caractère numérique.

Si des noms non valides sont fournis, la configuration échoue.

### Options de l'analyseur Regex

La seule option requise pour l'analyseur `regex` est l'option `format`.

L'option `debug` peut être utilisée lorsque des informations de débogage supplémentaires sont requises.

### Configuration

Pour créer un analyseur `regex`, utilisez `regex` comme `base_parser` et indiquez l'option `format`.

**Exemple : Exemples de configuration Regex****Exemple : Exemple d'analyse de journaux Apache**

L'exemple suivant peut être utilisé pour analyser 1234-5678-9123-4567 :

```
[parser|regex_parser]
base_parser=regex
format=(?<tag1>\d{4}) [-] (?<tag2>\d{4}) [-] (?<tag3>\d{4}) [-] (?<tag4>\d{4})
[filelog|some_info]
directory=D:\Logs
include=*.txt
parser=regex_parser
```

Les résultats indiquent :

```
tag1=1234
tag2=5678
tag3=9123
tag4=4567
```

Pour analyser des journaux Apache avec l'analyseur `regex`, fournissez le format `regex` spécifique aux journaux Apache :

```
[parser|regex_parser]
base_parser=regex
format=(?<remote_host>.* (?<remote_log_name>.* (?<remote_auth_user>.* \[(?<log_timestamp>.*)\] "(?<request>.*" (?<status_code>.* (?<response_size>.*
```

Les résultats indiquent :

```
127.0.0.1 - admin [10/Oct/2000:13:55:36 -0700] "GET /apache_pb.gif HTTP/1.0" 200 2326
remote_host=127.0.0.1
remote_log_name=-
remote_auth_user=admin
log_timestamp=10/Oct/2000:13:55:36 -0700
request=GET /apache_pb.gif HTTP/1.0
status_code=200
response_size=2326
```

Le code suivant montre un autre exemple d'analyse des journaux Apache.

```
[parser|regex_parser]
base_parser=regex
format=(?<remote_host>.* (?<remote_log_name>.* (?<remote_auth_user>.* \[(?<log_timestamp>.*)\] "(?<request>.* (?<resource>.* (?<protocol>.*))" (?<status_code>.* (?<response_size>.*
127.0.0.1 unknown - [17/Nov/2015:15:17:54 +0400] "\"GET /index.php HTTP/1.1\" 200 4868
remote_host=127.0.0.1 unknown
remote_log_name=unknown
remote_auth_user=-
log_timestamp=17/Nov/2015:15:17:54 +0400
request=GET /index.php HTTP/1.1
```

```
resource=/index.php
protocol=HTTP/1.1
status_code=200
response_size=4868
```

## Considérations de performances

L'analyseur `regex` consomme plus de ressources que d'autres analyseurs, par exemple l'analyseur `CLF`. Si vous pouvez analyser les journaux avec d'autres analyseurs, envisagez l'utilisation de ces derniers à la place de l'analyseur `regex` pour obtenir de meilleures performances.

Si aucun analyseur n'est fourni et que vous utilisez l'analyseur `regex`, définissez des formats les plus clairs possible. L'exemple suivant montre une configuration qui fournit des résultats plus performants. Cet exemple spécifie des champs ayant des valeurs numériques.

```
(?<remote_host>\d+\.\d+\.\d+\.\d+) (?<remote_log_name>.*) (?<remote_auth_user>.*) \[(?<log_timestamp>.*)\] "(?<request>.*)" (?<status_code>\d+) (?<response_size>\d+)
```

## Analyseur JSON

Vous pouvez personnaliser la configuration de l'analyseur JSON pour analyser de manière sélective le journal JSON.

Vous pouvez configurer des analyseurs CSV (Comma-Separated Value) pour les collecteurs **FileLog** et **WinLog**. Seuls les journaux JSON valides sont analysés avec l'analyseur JSON de l'agent Log Insight. Les analyseurs de journaux JSON non valides renvoient des résultats vides.

La configuration de l'analyseur JSON par défaut extrait tous les champs du journal JSON par l'agent Log Insight. Lorsque le journal JSON représente lui-même comme un objet JSON complexe, qui peut également contenir des objets JSON, l'analyseur utilise un trait de soulignement (`_`) pour concaténer les noms d'objets JSON imbriqués et hiérarchisés supérieurs. Ainsi les éléments correspondants portent un nom de champ informatif. Si le journal JSON contient également un tableau, les noms des éléments membres contiennent le nom du tableau suivi de l'index de l'élément dans le tableau.

L'analyseur JSON fournit également une option spécifique, appelée **champs**.

### Option « champs » de l'analyseur JSON

Vous pouvez utiliser l'option **champs** pour spécifier les champs à analyser dans la configuration. L'objectif de cette option est d'activer l'analyse sélective du journal JSON.

---

**Note** Pour l'analyse sélective, vous devez spécifier le chemin d'accès à l'élément JSON souhaité. Les objets JSON de différents niveaux doivent être séparés par un point (`.`).

---

La liste suivante fournit des exemples de configurations qui vous permettent d'analyser de manière sélective le journal JSON comme vous le souhaitez.

- Pour analyser plusieurs éléments du journal JSON, les éléments souhaités doivent être répertoriés comme paramètres pour l'option **champs** et séparés par des virgules. Voir l'exemple ci-dessous :

```
{ "operation" : { "timestamp" :
  "2018-11-22T15:28:58.094000", "thread_id" : "0x05673", "initiator" : "connector",
  "log_severity" : "info", "log_message" : "Requested connection to the server."},
  "operation_result" : "success" }
```

- Pour analyser uniquement les objets JSON les plus internes, tels que **timestamp**, **log\_severity** et **log\_message**, reportez-vous à l'exemple ci-dessous. Cet exemple de configuration produit les résultats de champ suivants : `operation_timestamp="2018-11-22T15:28:58.094000" and operation_log_severity="info"`

```
[parser|json_parser]
base_parser=json
fields=operation.timestamp,operation.log_severity, operation.log_message
```

- Pour analyser l'intégralité de l'objet JSON, incluez le chemin d'accès à l'objet suivi d'un astérisque (\*).

```
{ "product_name" : "LI Agent",
  "operation" : { "timestamp" : "2018-11-22T15:28:58.094000", "thread_id" :
  "0x05673", "initiator" : "connector", "log_severity" : "info", "log_message" :
  "Requested connection to the server."}, "operation_result" :
  "success" }
```

- Pour analyser uniquement l'objet **operation**, utilisez la configuration suivante :

```
[parser|json_parser]
base_parser=json
fields=operation.*
```

- Si le journal JSON contient un tableau et que vous voulez analyser uniquement des éléments spécifiques du tableau, utilisez l'index d'élément du tableau dans la configuration, comme indiqué dans cet exemple de configuration :

```
{
  "Records": [{
    "object": {
      "key": "/events/mykey",
      "size": 502,
      "eTag": "091820398091823",
      "sequencer": "1123123"
    }
  },
  {
    "object": {
```

```

        "key": "/events/user_key",
        "size": 128,
        "eTag": "09182039000001",
        "sequencer": "1123231"
    },
    {
        "object": {
            "key": "/events/admin_key",
            "size": 1024,
            "eTag": "09182039547241",
            "sequencer": "1123213"
        }
    }
]
}

```

- Pour analyser uniquement les éléments **key** et **size** du même journal, utilisez la configuration suivante pour générer les champs suivants :

```
records0_object_key="/events/mykey"
```

```
records0_object_size=502
```

```
records2_object_key="/events/admin_key"
```

```
records2_object_size=1024
```

```

[parser|json_parser]
base_parser=json
fields = Records0.object.key Records0.object.size, Records2.object.key,
Records2.object.size

```

- Pour analyser le champ **key** pour tous les éléments du tableau, utilisez la configuration suivante :

```

[parser|json_parser]
base_parser=json
fields=Records.#.object.key

```

- Pour analyser tous les champs, utilisez l'option `champs` avec un astérisque (\*). Cette configuration est équivalente à la configuration de l'analyseur JSON par défaut.

```

[parser|json_parser]
base_parser=json
fields=*

```

# Désinstallation des agents vRealize Log Insight

# 5

Si vous devez désinstaller un agent vRealize Log Insight, suivez les instructions appropriées au package de l'agent que vous avez installé.

Ce chapitre contient les rubriques suivantes :

- Désinstallez Log Insight Windows Agent
- Désinstaller le package RPM de Log Insight Linux Agent
- Désinstaller le module DEB de Log Insight Linux Agent
- Désinstaller le package bin de l'agent Linux Log Insight
- Désinstaller manuellement le package bin de l'agent Linux Log Insight

## Désinstallez Log Insight Windows Agent

Vous pouvez désinstaller Log Insight Windows Agent de l'écran Programmes et fonctionnalités du Panneau de configuration Windows.

### Conditions préalables

Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

### Procédure

- 1 Accédez à **Panneau de configuration > Programmes et fonctionnalités**.
- 2 Sélectionnez VMware vRealize Log Insight Windows Agent et cliquez sur **Désinstaller**.

### Résultats

Le programme de désinstallation arrête le service VMware vRealize Log Insight Windows Agent et supprime ses fichiers du système.

## Désinstaller le package RPM de Log Insight Linux Agent

Vous pouvez désinstaller le package RPM de Log Insight Linux Agent.

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console du terminal et exécutez `pgrep liagent` pour vérifier que Log Insight Linux Agent de VMware est installé et en cours d'exécution.

### Procédure

- ◆ Exécutez la commande suivante en remplaçant *VERSION* et *BUILD\_NUMBER* par la version et le numéro de build de l'agent installé.

```
rpm -e VMware-Log-Insight-Agent-VERSION-BUILD_NUMBER
```

### Résultats

Le programme de désinstallation arrête le démon de Log Insight Linux Agent de VMware et supprime tous ses fichiers à l'exception de ses propres journaux provenant du système.

## Désinstaller le module DEB de Log Insight Linux Agent

Vous pouvez désinstaller le module DEB de Log Insight Linux Agent.

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console du terminal et exécutez `pgrep liagent` pour vérifier que Log Insight Linux Agent de VMware est installé et en cours d'exécution.

### Procédure

- ◆ Exécutez la commande suivante.

```
dpkg -P vmware-log-insight-agent
```

### Résultats

Le programme de désinstallation arrête le démon de Log Insight Linux Agent de VMware et supprime tous ses fichiers à l'exception de ses propres journaux provenant du système.

## Désinstaller le package bin de l'agent Linux Log Insight

Vous pouvez désinstaller le package `.bin` de Log Insight Linux Agent avec un script vRealize Log Insight.

### Conditions préalables

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.

- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console de terminal et exécutez la commande `pgrep liagent` pour vérifier que Log Insight Linux Agent VMware vRealize est installé et qu'il s'exécute.

### Procédure

- 1 À l'invite du shell, entrez la commande suivante pour démarrer le script.

```
loginsight-agent-uninstall
```

- 2 Vous pouvez vérifier que la désinstallation s'est terminée correctement en vérifiant que le code d'erreur renvoyé par la commande suivante est 0.

```
echo $?
```

## Désinstaller manuellement le package bin de l'agent Linux Log Insight

Vous pouvez désinstaller le package .bin de Log Insight Linux Agent manuellement si vous choisissez de ne pas utiliser le script de désinstallation.

### Conditions préalables

#### Désinstaller manuellement le package bin de l'agent Linux Log Insight

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console de terminal et exécutez la commande `pgrep liagent` pour vérifier que Log Insight Linux Agent VMware vRealize est installé et qu'il s'exécute.

### Procédure

- 1 Interrompez le démon de Log Insight Linux Agent en exécutant la commande suivante `sudo service liagentd stop` ou `sudo /sbin/service liagentd stop` pour les distributions Linux plus anciennes.
- 2 Supprimer manuellement les fichiers de Log Insight Linux Agent
  - `/usr/lib/loginsight-agent` - répertoire des fichiers de licence et binaires de démon.
  - `/usr/bin/loginsight-agent-support` - utilisé pour générer le bundle de support de Log Insight Linux Agent.
  - `/var/lib/loginsight-agent` - répertoire de stockage de la base de données et des fichiers de configuration.
  - `/var/log/loginsight-agent` - répertoire de journaux pour Log Insight Linux Agent.
  - `/var/run/liagent/liagent.pid` - fichier PID de Log Insight Linux Agent. Si le fichier n'est pas supprimé automatiquement, supprimez-le manuellement.

- `/etc/init.d/liagentd` - répertoire de script du démon Log Insight Linux Agent.
- `/usr/lib/systemd/system/liagentd.service`

# Dépannage des agents vRealize Log Insight

# 6

Les informations de dépannage connues peuvent vous aider à diagnostiquer et à corriger les problèmes liés à l'utilisation des agents vRealize Log Insight.

Ce chapitre contient les rubriques suivantes :

- Créer un bundle de support pour Log Insight Windows Agent
- Créer un bundle de support pour Log Insight Linux Agent
- Définir le niveau de détail des journaux dans le Log Insight Agents
- L'interface utilisateur d'administration n'affiche pas Log Insight Agents
- Les agents vRealize Log Insight n'envoient pas d'événements
- Ajouter une règle d'exception de trafic sortant pour Log Insight Windows Agent
- Autoriser les connexions sortantes à partir de Log Insight Windows Agent dans un Pare-feu Windows
- Le déploiement en masse de Log Insight Windows Agent a échoué
- Les agents Log Insight Agents rejettent les certificats auto-signés
- Le serveur vRealize Log Insight rejette la connexion pour trafic non chiffré

## Créer un bundle de support pour Log Insight Windows Agent

Si Log Insight Windows Agent ne fonctionne pas comme prévu en raison d'un problème, vous pouvez envoyer une copie des fichiers de configuration et du journal au support VMware.

### Procédure

- 1 Connectez-vous à la machine cible sur laquelle vous avez installé Log Insight Windows Agent.
- 2 Cliquez sur le bouton **Démarrer** de Windows, puis sur **VMware > Log Insight Agent - Collect support Bundle**.

- 3 (Facultatif) Si le raccourci n'est pas disponible, accédez au répertoire d'installation de Log Insight Windows Agent et double-cliquez sur `loginsight-agent-support.exe`.

---

**Note** Le répertoire d'installation par défaut est `C:\Program Files (x86)\VMware\Log Insight Agent`

---

#### Résultats

Le bundle est généré et enregistré sous la forme d'un fichier `.zip` dans le répertoire `Mes documents`.

#### Étape suivante

Transférez le bundle de support au support VMware, comme requis.

## Créer un bundle de support pour Log Insight Linux Agent

Si Log Insight Linux Agent ne fonctionne pas comme prévu en raison d'un problème, vous pouvez envoyer une copie des fichiers de configuration et du journal au support VMware.

#### Procédure

- 1 Connectez-vous à la machine cible sur laquelle vous avez installé Log Insight Linux Agent.
- 2 Exécutez la commande suivante.

```
/usr/lib/loginsight-agent/bin/loginsight-agent-support
```

#### Résultats

Ce bundle est généré et enregistré en tant que fichier `.zip` dans le répertoire actuel.

#### Étape suivante

Transférez le bundle de support au support VMware, comme requis.

## Définir le niveau de détail des journaux dans le Log Insight Agents

Vous pouvez modifier le fichier de configuration de l'agent vRealize Log Insight pour changer de niveau de journalisation.

#### Conditions préalables

Pour Log Insight Linux Agent :

- Connectez-vous en tant que **root** ou utilisez `sudo` pour exécuter des commandes de console.
- Connectez-vous à la machine Linux sur laquelle vous avez installé Log Insight Linux Agent, ouvrez une console et exécutez `pgrep liagent` pour vérifier que VMware vRealize Log Insight Linux Agent est installé et fonctionne.

Pour Log Insight Windows Agent :

- Connectez-vous à la machine Windows sur laquelle vous avez installé vRealize Log Insight Windows Agent et démarrez le Gestionnaire de services pour vérifier que le service vRealize Log Insight Agent est installé.

#### Procédure

- 1 Accédez au dossier contenant le fichier `liagent.ini`.

Système d'exploitation	Chemin
Linux	<code>/var/lib/loginsight-agent/</code>
Windows	<code>%ProgramData%\VMware\Log Insight Agent</code>

- 2 Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.
- 3 Modifiez le niveau de débogage de journal dans la section `[logging]` du fichier `liagent.ini`.

**Note** Plus le niveau de débogage est élevé, plus il a d'impact sur l'agent vRealize Log Insight. La valeur par défaut est 0. C'est la valeur recommandée. Le niveau de débogage 1 permet d'obtenir plus d'informations. Il est recommandé pour le dépannage de la plupart des problèmes. Le niveau de débogage 2 permet d'obtenir des informations détaillées. Utilisez les niveaux 1 et 2 uniquement lorsque le support VMware vous le demande.

```
[logging]
; The level of debug messages to enable: 0..2
debug_level=1
```

- 4 Enregistrez et fermez le fichier `liagent.ini`.

#### Résultats

Le niveau de débogage des journaux a changé.

## L'interface utilisateur d'administration n'affiche pas Log Insight Agents

Aucune information sur les instances de Log Insight Agents ne s'affiche dans la page Agents de l'interface utilisateur d'administration.

#### Problème

Après l'installation de Log Insight Agents, vous ne voyez pas Log Insight Agents dans la page Agents de l'interface utilisateur d'administration.

#### Cause

Cet incident est le plus souvent dû à des problèmes de connectivité réseau ou à une configuration incorrecte de Log Insight Agents dans le fichier `liagent.ini`.

### Solution

- ◆ Vérifiez que le système Windows ou Linux sur lequel les agents Log Insight Agents sont installés dispose d'une connectivité au serveur vRealize Log Insight.

- ◆ Vérifiez que Log Insight Agents utilise le protocole cfapi.

Lors de l'utilisation du protocole syslog, l'interface utilisateur n'affiche pas Log Insight Windows Agents.

- ◆ Affichez le contenu des fichiers journaux de Log Insight Agents se trouvant dans les répertoires suivants.

- Windows - %ProgramData%\VMware\Log Insight Agent\log

- Linux - /var/log/loginsight-agent/

Recherchez les messages de journaux qui contiennent les phrases `Config transport error: Couldn't resolve host name et Resolver failed. No such host is known.`

- ◆ Vérifiez que le fichier `liagent.ini` contient la configuration appropriée pour le serveur vRealize Log Insight cible. Voir [Définir le serveur vRealize Log Insight cible](#) et [Spécifier la cible d'un agent](#).

## Les agents vRealize Log Insight n'envoient pas d'événements

Une configuration incorrecte peut empêcher les agents vRealize Log Insight de transférer les événements vers le serveur vRealize Log Insight. Si un canal de collecte de fichiers plats n'est pas correctement configuré, des messages de ce type peuvent s'afficher : "Des paramètres incorrects ont été obtenus pour le canal 'CHANNEL\_NAME'. Le canal 'CHANNEL\_NAME' restera dormant jusqu'à ce qu'il soit correctement configuré".

### Problème

Les instances des agents vRealize Log Insight apparaissent sur la page **Administration > Agent**, mais aucun événement provenant des noms d'hôte des agents vRealize Log Insight n'apparaît sur la page Analyse interactive. Le canal de collecte de fichiers plats n'est pas correctement configuré.

### Cause

Une configuration incorrecte peut empêcher les agents vRealize Log Insight de transférer les événements vers le serveur vRealize Log Insight.

### Solution

- ◆ Définissez un canal de collecte valide. Vérifiez que le canal de collecte de fichiers plats est correctement configuré. Reportez-vous à [Chapitre 4 Configuration des agents vRealize Log Insight](#).

- ◆ Pour l'agent vRealize Log Insight Windows, suivez ces étapes.
  - Si des canaux Windows sont activés, affichez le contenu des fichiers journaux de l'agent vRealize Log Insight Windows situés dans `%ProgramData%\VMware\Log Insight Agent\log`. Recherchez les messages de journaux liés à la configuration des canaux qui contiennent le texte `Abonné au canal CHANNEL_NAME`. Les canaux généralement utilisés sont `Application`, `System` et `Security`.
  - Si un canal n'est pas configuré correctement, vous risquez de voir des messages de journaux semblables à `Abonnement impossible aux événements du canal CHANNEL_NAME. Code d'erreur : 15007. Le canal spécifié est introuvable. Vérifiez la configuration du canal.. Il se peut qu'un code d'erreur différent de 15007 soit affiché.`
  - Si un canal de collecte de fichiers plats n'est pas correctement configuré, des messages de ce type peuvent s'afficher : `Des paramètres incorrects ont été obtenus pour le canal 'CHANNEL_NAME'. Le canal 'CHANNEL_NAME' restera dormant jusqu'à ce qu'il soit correctement configuré.`
- ◆ Pour les agents vRealize Log Insight Windows et vRealize Log Insight Linux, suivez ces étapes.
  - ◆ Si aucun canal de collecte de fichiers plats n'est configuré, vous risquez de voir des messages semblables à `Impossible de trouver la section 'filelog' dans la configuration. Le collecteur de journaux de fichiers plats restera dormant jusqu'à ce qu'il soit correctement configuré.`

Le contenu des fichiers journaux des agents vRealize Log Insight se trouve dans les répertoires suivants.

    - Windows - `%ProgramData%\VMware\Log Insight Agent\log`
    - Linux - `/var/log/loginsight-agent/`

#### Étape suivante

Pour plus d'informations sur la configuration des agents vRealize Log Insight, reportez-vous à [Configurer Log Insight Windows Agent](#) et [Configurer Log Insight Linux Agent](#).

## Ajouter une règle d'exception de trafic sortant pour Log Insight Windows Agent

Définissez une règle d'exception pour débloquer Log Insight Windows Agent dans le Pare-feu Windows.

La procédure s'applique à Windows Server 2008 R2 et version ultérieure, ainsi qu'à Windows 7 et version ultérieure.

### Conditions préalables

- Vérifiez que vous disposez d'un compte administrateur ou d'un compte doté de privilèges administratifs.

### Procédure

- 1 Sélectionnez **Démarrer > Exécuter**.
- 2 Tapez `wf.msc` et cliquez sur **OK**.
- 3 Cliquez avec le bouton droit sur **Règles de trafic sortant** dans le volet de gauche et cliquez sur **Nouvelle règle**.
- 4 Sélectionnez **Personnalisé** et suivez les instructions de l'assistant pour définir les options ci-dessous.

Option	Description
Programme	liwinsvc.exe
Service	LogInsightAgentService
Protocole et ports	TCP 9000 pour cfapi et 514 pour syslog

- 5 Sur la page Spécifier les profils pour lesquels cette règle s'applique, sélectionnez le type de réseau approprié.
  - Domaine
  - Public
  - Privée

---

**Note** Vous pouvez sélectionner tous les types de réseaux pour vous assurer que la règle d'exception est active quel que soit le type de réseau.

---

### Étape suivante

Accédez au répertoire des journaux de Log Insight Windows Agent `%ProgramData%\VMware\Log Insight Agent\log` et ouvrez le dernier fichier de journal. Si des événements récents contiennent les messages `Erreur de transport de configuration : impossible de résoudre le nom d'hôte et Le programme de résolution a échoué. Hôte inconnu`, redémarrez le service Log Insight Windows Agent et la machine Windows.

---

**Note** La reconnexion entre le service Log Insight Windows Agent et le serveur peut prendre jusqu'à 5 minutes.

---

## Autoriser les connexions sortantes à partir de Log Insight Windows Agent dans un Pare-feu Windows

Configurez les paramètres de Pare-feu Windows de manière à autoriser les connexions sortantes du Log Insight Windows Agent vers le serveur vRealize Log Insight.

Une fois le service Log Insight Windows Agent installé et démarré, le pare-feu local ou le domaine Windows peuvent limiter la connectivité avec le serveur vRealize Log Insight cible.

La procédure s'applique à Windows Server 2008 R2 et version ultérieure, ainsi qu'à Windows 7 et version ultérieure.

### Conditions préalables

- Vérifiez que vous disposez d'un compte administrateur ou d'un compte doté de privilèges administratifs.

### Procédure

- 1 Sélectionnez **Démarrer > Exécuter**.
- 2 Tapez `wf.msc` et cliquez sur **OK**.
- 3 Dans le volet Actions, cliquez sur **Propriétés**.
- 4 Dans l'onglet **Profil de domaine**, sélectionnez **Autoriser (par défaut)** dans le menu déroulant **Connexions sortantes**.

Si l'ordinateur n'est pas connecté à un domaine, vous pouvez sélectionner **Profil privé** ou **Profil public**, selon le type de réseau auquel l'ordinateur est connecté.

- 5 Cliquez sur **OK**.

### Étape suivante

Définissez une règle d'exception de déblocage pour Log Insight Windows Agent dans le Pare-feu Windows. Reportez-vous à [Ajouter une règle d'exception de trafic sortant pour Log Insight Windows Agent](#).

## Le déploiement en masse de Log Insight Windows Agent a échoué

Le déploiement en masse de Log Insight Windows Agent a échoué sur les machines cibles.

### Problème

Après un déploiement en masse sur les machines d'un domaine Windows à l'aide d'objets de stratégie de groupe, Log Insight Windows Agent ne parvient pas à s'installer en tant que service local.

### Cause

Il est possible que les paramètres de stratégie de groupe empêchent l'installation correcte de Log Insight Windows Agent.

### Solution

- 1 Modifiez les paramètres d'objet de stratégie de groupe et redéployez le Log Insight Windows Agent.
  - a Cliquez avec le bouton droit sur l'objet de stratégie de groupe, sélectionnez **Modifier** et accédez à **Configuration ordinateur > Stratégies > Modèles d'administration > Système > Ouverture de session**.
  - b Activez la stratégie **Toujours attendre le réseau lors du démarrage de l'ordinateur et de l'ouverture de session**.
  - c Accédez à **Configuration ordinateur > Stratégies > Modèles d'administration > Système > Stratégie de groupe**.
  - d Activez la stratégie **Temps d'attente de traitement de stratégie de démarrage** et définissez **Temps d'attente (en secondes)** sur 120.
- 2 Exécutez la commande `gpupdate /force /boot` sur les machines cibles.

## Les agents Log Insight Agents rejettent les certificats auto-signés

Les agents Log Insight Agents rejettent un certificat auto-signé.

### Problème

Un agent vRealize Log Insight rejette le certificat auto-signé et ne peut pas établir une connexion avec le serveur.

---

**Note** Si vous rencontrez des problèmes de connexion avec l'agent, vous pouvez générer des journaux détaillés et faire une vérification en définissant le niveau de débogage pour l'agent sur 1. Pour plus d'informations, consultez [Définir le niveau de détail des journaux dans le Log Insight Agents](#).

---

### Cause

Les messages affichés dans le journal de l'agent ont des causes spécifiques.

Message	Cause
Rejet d'un certificat auto-signé par un pair. La clé publique ne correspond pas à la clé du certificat enregistrée précédemment.	<ul style="list-style-type: none"> <li>■ Cela peut se produire lorsque le certificat vRealize Log Insight est remplacé.</li> <li>■ Cela peut se produire si la haute disponibilité (HA) activée dans l'environnement de cluster est configurée avec des certificats auto-signés différents sur les nœuds vRealize Log Insight.</li> </ul>
Rejet d'un certificat auto-signé par un pair. Vous devez disposer d'un certificat reçu précédemment, signé par une autorité de certification de confiance.	Un certificat signé par une autorité de certification de confiance est enregistré côté agent.

### Solution

- ◆ Vérifiez que votre nom d'hôte cible est une instance approuvée de vRealize Log Insight puis supprimez manuellement le certificat précédent du répertoire vRealize Log Insight Agent cert.
  - Pour Log Insight Windows Agent, accédez à `C:\ProgramData\VMware\Log Insight Agent\cert`.
  - Pour Log Insight Linux Agent, accédez à `/var/lib/loginsight-agent/cert`.

**Note** Certaines plates-formes peuvent utiliser des chemins d'accès non standard pour l'enregistrement des certificats approuvés. Log Insight Agents dispose d'une option pour configurer le chemin d'accès vers le magasin des certificats approuvés en configurant le paramètre de configuration `ssl_ca_path=<fullpath>`. Remplacez `<fullpath>` par le chemin d'accès vers le fichier groupé des certificats racines approuvés. Reportez-vous à la rubrique [Configurer les paramètres SSL Log Insight Agents](#).

## Le serveur vRealize Log Insight rejette la connexion pour trafic non chiffré

Le serveur vRealize Log Insight rejette la connexion avec Log Insight Agents lorsque vous tentez d'envoyer du trafic non chiffré.

Vous pouvez configurer le serveur vRealize Log Insight de manière qu'il accepte les connexions non-SSL ou configurer les agents Log Insight Agents pour qu'ils envoient des données par l'intermédiaire du protocole de connexion SSL `cfapi`.

### Problème

Lorsque vous tentez d'utiliser `cfapi` pour envoyer du trafic non chiffré, le serveur vRealize Log Insight rejette votre connexion. Un des messages d'erreur suivants s'affiche dans le journal de l'agent : `403 Forbidden` ou `403 Only SSL connections are allowed`.

## Cause

vRealize Log Insight est configuré pour accepter uniquement les connexions SSL, mais les instances de Log Insight Agents sont configurées pour utiliser une connexion non-SSL.

## Solution

- 1 Configurez le serveur vRealize Log Insight de manière qu'il accepte la connexion non-SSL.
  - a Accédez à l'onglet **Administration**.
  - b Dans Configuration, cliquez sur **SSL**.
  - c Sous l'en-tête API pour serveur SSL, désélectionnez **Exiger une connexion SSL**.
  - d Cliquez sur **Enregistrer**.
- 2 Configurez l'agent vRealize Log Insight de manière qu'il envoie les données par l'intermédiaire du protocole de connexion SSLcfapi.
  - a Accédez au dossier contenant le fichier `liagent.ini`.

Système d'exploitation	Chemin
Linux	<code>/var/lib/loginsight-agent/</code>
Windows	<code>%ProgramData%\VMware\Log Insight Agent</code>

- b Ouvrez le fichier `liagent.ini` dans un éditeur de texte quelconque.
- c Remplacez la valeur de la clé `ssl` dans la section `[server]` du fichier `liagent.ini` par « yes » et celle du protocole par `cfapi`.

```
proto=cfapi
ssl=yes
```

- d Enregistrez et fermez le fichier `liagent.ini`.