shinken-scheduler-export-data - export des données du Scheduler

Sommaire

- Intérêt global de la commande
- Fonctionnement général de la commande
  - Récupération des données des checks auprès des schedulers
  - Options de la commande
  - Options de la commande pour les extractions de types sizing-pollers et identify-most-consuming-checks
  - Résultat du lancement sur le terminal
    - Le retour du terminal
    - CSV des données
  - Contenu du fichier csv généré
  - Chargement et analyse des données
    - Chargement du fichier csv
    - Analyse des données
- Erreurs possibles
  - La commande est lancée depuis un serveur autre que l'Arbiter
  - Problème de paramétrage du mot de passe d'accès (daemon__export_data__password dans /etc/shinken/shinken.cfg)
    - Le mot de passe est vide ou non configuré dans le fichier /etc/shinken/shinken.cfg
    - Le mot de passe fourni par l'utilisateur est incorrect
  - Problème de paramétrage de la désactivation de l'export des données des démons (scheduler__export_data__enabled parameter dans le .cfg du Scheduler)
  - Problèmes avec le réseaux ou l'état du Scheduler
    - Le Scheduler n'est pas accessible (éteint ou problème réseau)
    - Le Scheduler n'a pas encore reçu de configuration de l'Arbiter
  - Passer les erreurs et travailler avec des données partielles

Intérêt global de la commande

Sur les grands environnements, la majorité de la charge CPU va être consommée par les sondes de supervision. Quand elles sont nombreuses et qu'elles sont ordonnancées avec des intervalles de temps différents, il peut être difficile :

- d'extrapoler, par rapport au nombre d'hôtes déjà présents, de combien de CPUs sur ses Pollers une infrastructure à besoin.
- de savoir, sur une infrastructure, qu'elles sont les sondes les plus consommatrices en termes de CPU :
  - une fois identifiées, il est intéressant de passer du temps pour optimiser ces sondes afin de diminuer le nombre de CPU nécessaires.

C'est pour répondre à ces problématiques que nous avons crée la commande **shinken-scheduler-export-data**.

Vous trouverez des exemples d'analyses de résultat de la commande dans les pages suivantes :

- Vous avez un environnement avec un certain nombre d'hôtes, vous souhaitez dimensionner vos Pollers si vous rajoutez 10, 20 ou 50% d'hôtes en plus :
  - dans ce cas, il faut se référer à la page Dimensionnement des Pollers à l'aide de la commande shinken-scheduler-export-data.
- Si vous souhaitez diminuer le nombre de CPU de vos Pollers en identifiant, et optimisant, les sondes les plus consommatrices :
  - dans ce cas, il faut se référer à la page Identification des checks les plus consommateurs à l'aide de la commande shinken-scheduler-export-data.

Cette commande interroge les Schedulers, car ce sont eux qui centralisent les informations de temps d'exécution des sondes de supervision exécutées par les Pollers.

- Le fichier de résultat est un fichier .csv qui contient une ligne par check exécuté par les Pollers, avec suivant les options des données différentes.
  - Ces données seront les données constatées par les Pollers.
- Vous pouvez aussi simuler la génération des lignes de check sur une période.
  - L'intérêt est que si certains checks sont exécutés toutes les heures, ils n'auront pas le même poids dans vos besoins en ressource qu'un check exécuté toutes les minutes.

Fonctionnement général de la commande

Récupération des données des checks auprès des schedulers

La commande "shinken-scheduler-export-data" est seulement utilisable depuis le serveur de l'**Arbiter**.

- En effet, elle se connecte sur tous les Schedulers, et c'est le seul démon habilité à cela.

La commande va lire la configuration depuis /etc/shinken et va pouvoir se connecter sur tous les Schedulers qui sont définis dans l'architecture. Le fichier de résultat peut être :

- soit ANONYME (*aucun nom d'hôte, royaume, etc, ne sera présent dans le fichier généré, voir l'option* ***--anonymous***),
  - L'extraction sera anonymisée (*seule un hash des noms sera lisible*)
- soit NORMAL, donc avec les noms des éléments (*par défaut*)
  - Un mot de passe vous sera systématiquement demandé pour générer ce fichier
  - Le mot de passe est celui défini dans le fichier de configuration de Shinken (*/etc/shinken/shinken.cfg*) dans la clé : **daemon__export_data__password** (*voir la page* Fichier de configuration (shinken.cfg)).

Il est possible de désactiver l'extraction pour un Scheduler via l'option définie dans son fichier de configuration avec le paramètre : **scheduler__export_data__enabled** (*voir la page* Le Scheduler).

Options de la commande

La commande accepte les options suivantes :

Nom	Type	Unité	Défaut	Commentaire
`--export-type`	Texte	--	--	Cette option permet de choisir quel type de donnée sera exportées. Les valeurs possibles de cette option sont : • **sizing-pollers**: Export les données pour le dimensionnement des Poller. • **identify-most-consuming-checks**: Export les données pour identifier les sondes les plus consommatrices. `--export-type=sizing-pollers`
`--anonymous`	--	--	--	Cette option permet d'avoir un export anonyme, c'est-à-dire que les noms des éléments ne seront pas présents. • Cela vous permettra d'exporter les données dans les situations où la confidentialité est nécessaire. • Par exemple, mettre à disposition ces données a votre support Shinken sans que les noms de vos machines ne soient diffusés `--anonymous`
`--scheduler-timeout`	Entier	sec	10	Permets de configurer le temps d'attente acceptée sur les appels des Schedulers avant de retourner une erreur. `--scheduler--timeout=30`
`--realm-filter`	Texte	--	--	Prends le nom du royaume que l'on souhaite utiliser comme filtre afin de n'avoir que des Schedulers de ce royaume dans notre extraction de données. (*on ne peut mettre qu'un seul royaume dans le filtre*) `--realm-filter=France`
`--dont-stop-on-scheduler-error`	Booléen	--	--	Si activé, la commande ne s'arrêtera plus si un ou plusieurs Scheduler est éteint/mal configuré, ou s'il ne répond pas (*timeout*). A noter que dans ce cas les données extraites risquent d'être partielles avec des éléments manquants. De plus, les Schedulers spare inactifs sont automatiquement passés, il n'est pas nécessaire d'utiliser cette option pour ces derniers. `--dont-stop-on-scheduler-error`

`--debug`	--	--	--	Option permettant d'avoir plus d'affichage sur le fonctionnement interne de la commande. **Pour le support uniquement.** `--debug`

Options de la commande pour les extractions de types sizing-pollers et identify-most-consuming-checks

Nom	Type	Unité	Défaut	Commentaire
`--simulate-scheduling-for-X-seconds`	Entier	sec	**0**	Cette option permet, de simuler une activité normale des checks, en rajoutant leurs données après chaque intervalle de vérification spécifique à chaque check. • Une ligne de donnée sera donc ajoutée pour chaque exécution simulée. • **Par exemple**: avec simulate-scheduling-for-X-seconds=3600, un check avec un intervalle de vérification à une minute générera 60 lignes. Cette simulation ne prend pas en compte : • les **p ériodes de vérification** (*prévu pour une futur version*), • les **intervalles de nouvelles tentatives de confirmations d'état** (*retry_interval*) => Difficile de définir de manière pertinente combien d'incidents futurs pourraient apparaître et engendrer une activité supplémentaire sur les Pollers à cause de ces sur-vérifications de confirmation. Donc les valeurs données par cette simulation ne seront pas forcément parfaitement alignées avec la réalité du futur. `--simulate-scheduling-for-X-seconds=3600`

Résultat du lancement sur le terminal

Le lancement de la commande fera deux choses :

- un retour dans le terminal pour voir le retour global de la commande,
- la génération d'un fichier .csv avec l'ensemble des données.
  - Les lignes du fichier csv représente chaque occurrence d'exécution de chaque check.

Deux exemples seront fournis pour illustrer la documentation:

- Exemple 1 : Exécution sécurisée des données pour le dimensionnement des Pollers (*--export-type=sizing-pollers*) et sans simulation dans le futur.

```
shinken-scheduler-export-data --export-type=sizing-pollers
```

- Exemple 2 : Exécution des données pour identifier les sondes qui consomme le plus de CPU sur les Pollers (*--export-type=identify-most-consuming-checks*) avec une demande de simulation d'une heure supplémentaire (*--simulate-scheduling-for-X-seconds=3600*) , de manière anonyme (*--anonymous*) pour ne pas avoir les noms des hôtes.

```
shinken-scheduler-export-data --export-type=identify-most-consuming-checks --simulate-scheduling-for-X-seconds=3600 --anonymous
```

Le retour du terminal

Le retour donne :

- la liste des Schedulers,
- l'adresse du fichier où le CSV a été généré,
- le nombre de lignes générées (*qui correspond aux nombres d'éléments et simulé si l'option **--simulate-scheduling-for-X-seconds** a été utilisé)*

Exemple 1

```
~#shinken-scheduler-export-data --export-type=sizing-pollers
Enter export data password ("daemon__export_data__password" in file /etc/shinken/shinken.cfg):

_____________  Get check dumps from schedulers (export wih names) _____________
  All >
    - scheduler-spare               : SKIPPED (sleeping spare)
    - scheduler-secondary           : OK
    - scheduler-master              : OK
_____________  [ sizing-pollers ] Writing to CSV file: /tmp/dump-schedulers--sizing-pollers--no-simulate-scheduling--with-names--2023-04-24-10h04m05s.csv _____________
  Wrote 26 lines in 0.001s
```

Exemple 2

```
~#shinken-scheduler-export-data --export-type=identify-most-consuming-checks --anonymous --simulate-scheduling-for-X-seconds=3600
_____________  Get check dumps from schedulers (anonymous dump) _____________
  All >
    - scheduler-spare               : SKIPPED (sleeping spare)
    - scheduler-secondary           : OK
    - scheduler-master              : OK
_____________  [ identify-most-consuming-checks ] Writing to CSV file: /tmp/dump-schedulers--identify-most-consuming-checks--simulate-scheduling-for-3600-seconds--anonymous--2023-04-24-
h05m29s.csv _____________
  Wrote 530 (with 504 simulated scheduling) lines in 0.008s
```

À noter que dans l'exemple 2, le nombre de lignes est bien plus élevées à causes des nombreuses lignes de la simulation (*530 au lieu de 26 dans notre exemple*).

CSV des données

Elle va générer un fichier **csv** dans le répertoire **/tmp** du serveur dont le nom va dépendre des options utilisées :

- Exemple 1 :

  /tmp/dump-schedulers--sizing-pollers--no-simulate-scheduling--with-names--2023-04-24-10h04m05s.csv

- Exemple 2 :

  /tmp/dump-schedulers--identify-most-consuming-checks--simulate-scheduling-for-3600-seconds--anonymous--2023-04-24-10h05m29s.csv

Contenu du fichier csv généré

Dans le fichier CSV, on a les données suivantes :

host_uuid	**UUID** de l'hôte.
host_name	Nom de l'hôte (Vide *si l'option* ***--anonymous*** *est défini dans la commande*).
check_uuid	**UUID** du check.
check_name	Nom du check (Vide *si l'option* ***--anonymous*** *est défini dans la commande*); • vide si la vérification porte sur un hôte
command_name	Nom de la commande (Vide *si l'option* ***--anonymous*** *est défini dans la commande*) .
command_name_anonymous_hash	Hash du nom pour l'identifier sur un export avec l'option *--anonymous*.
realm	Nom du royaume (Vide *si l'option* ***--anonymous*** *est défini dans la commande*) .
realm_anonymous_hash	Hash du nom du royaume pour l'identifier sur un export avec l'option *--anonymous*.
check_interval	Intervalle entre les vérifications.
retry_interval	Intervalle de nouvelles tentatives de confirmations d'état.

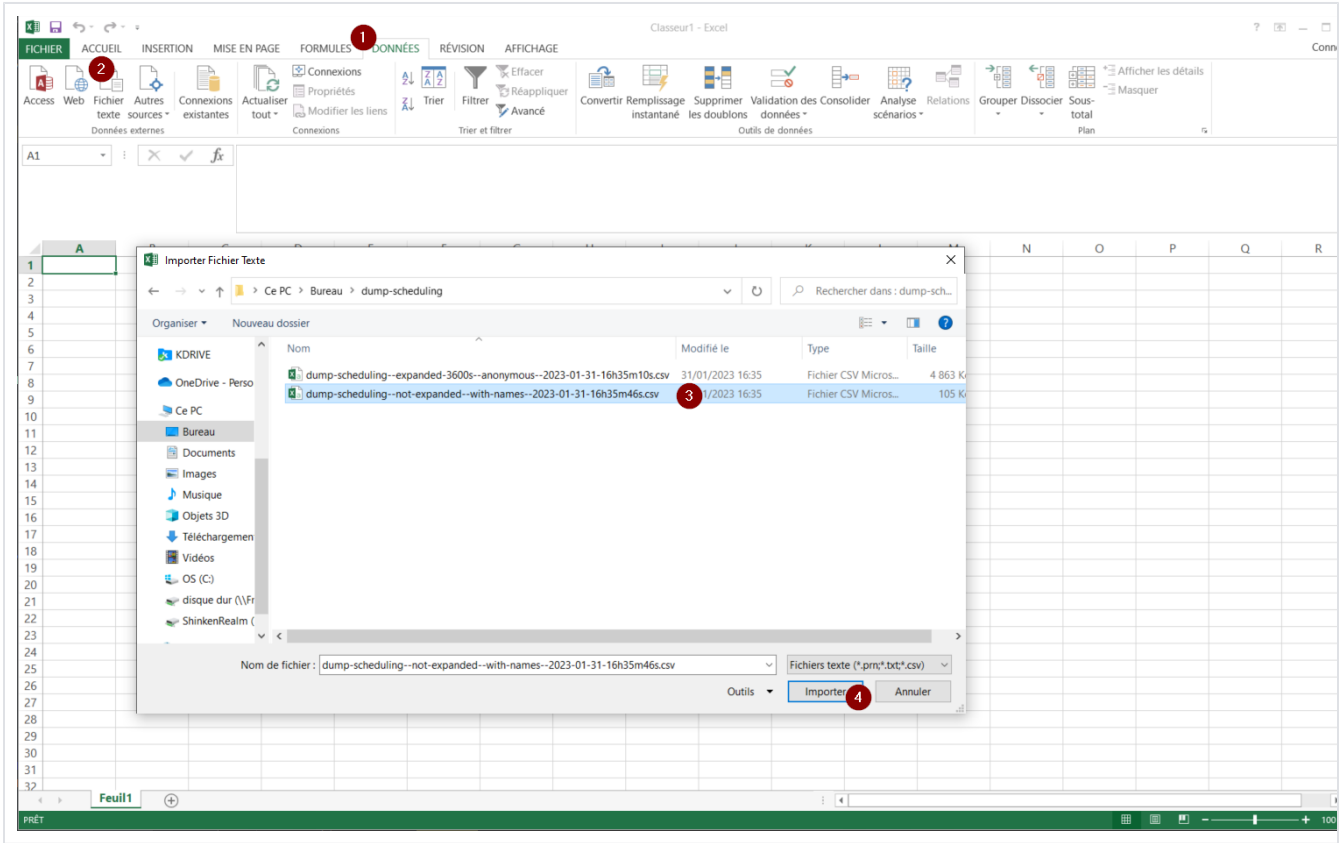
cpu_time	Temps CPU utilisé par le check : • Format chaine de caractère représentant un nombre flottant (*float*) avec un "." • À bien faire attention quand on charge dans un Excel français, il faut changer le séparateur de "," à "."
scheduled_epoch	Seconde où le check est planifié (*en epoch, depuis le 1er janvier 1970*).
scheduled_date	Format *YYYY-MM-DD hh:mm:ss*
is_simulated	Booléen permettant de savoir si le check est issu d'une vraie exécution prévue dans le Scheduler, ou s'il est issu d'une simulation du check dans le futur (*via l'option **--simulate-scheduling-for-X-seconds***)
last_reference_epoch	Seconde où le check s'est exécuté pour la dernière fois et que son état a été confirmé (*"HARD"*) (*en epoch, depuis le 1er janvier 1970*).

Chargement et analyse des données

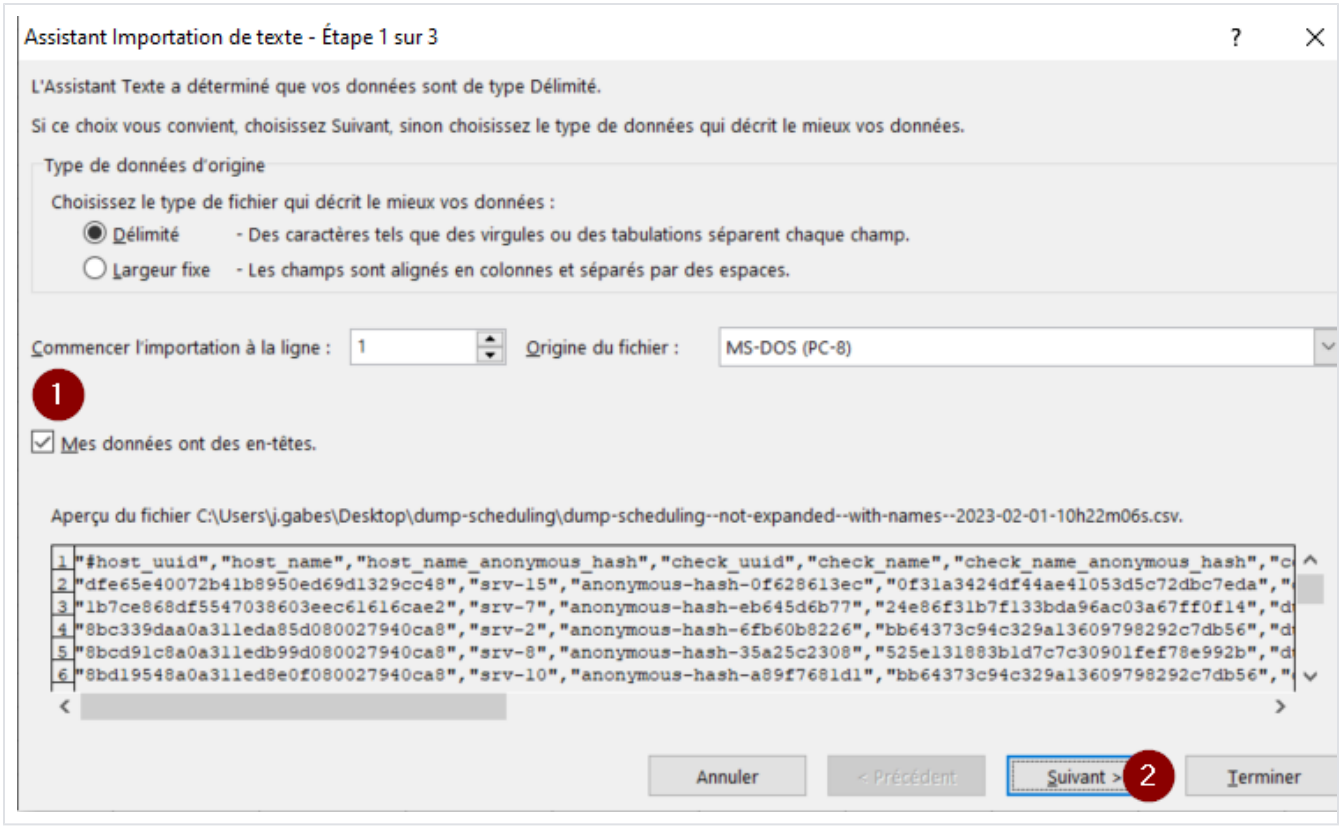
Chargement du fichier csv

Les fichiers CSV générés peuvent être importés dans un tableur comme Excel.

- L'import dans ce dernier se fait simplement, dans les autres tableurs l'import suivra un mécanisme similaire.
- Depuis un tableur vide, il faut passer par **DonnéesFichier texte**.

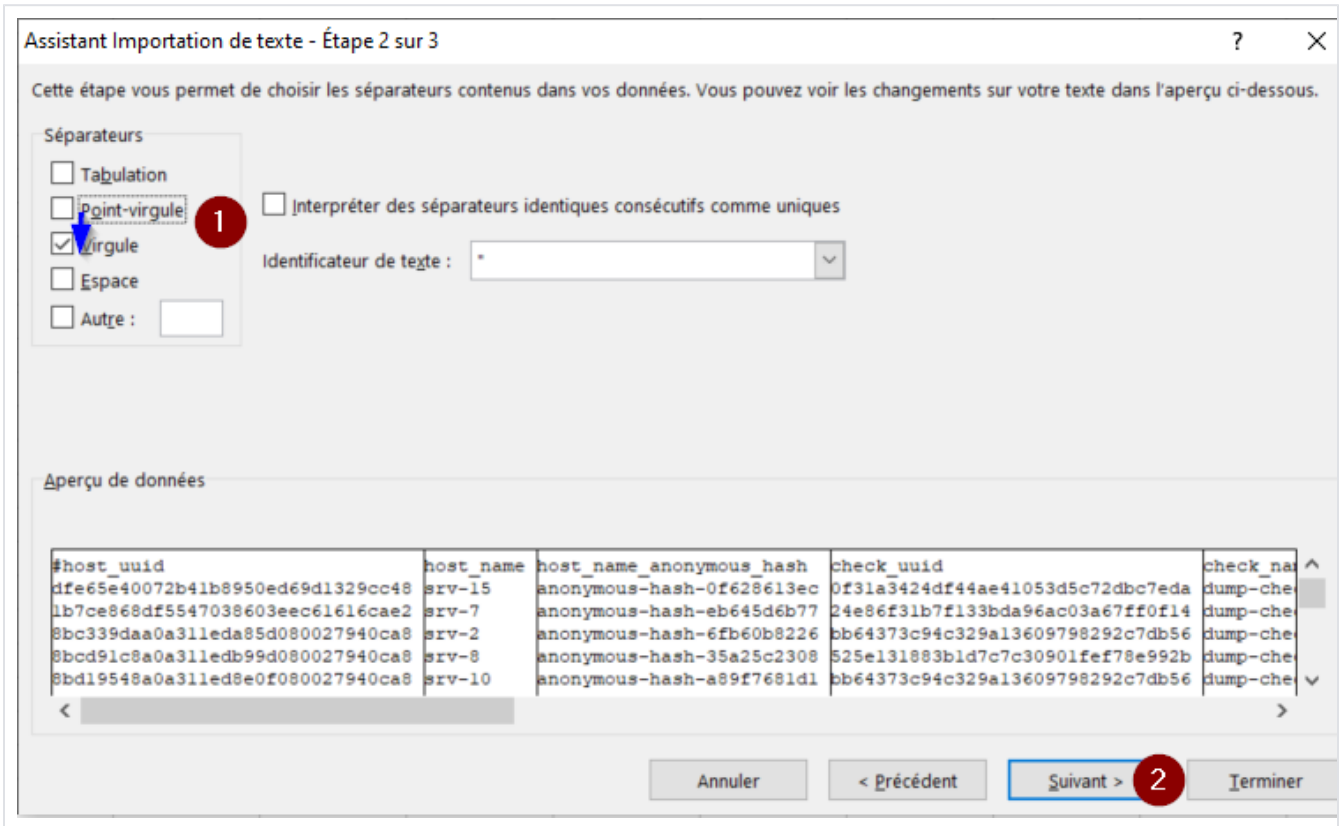


Vu que le fichier a une première ligne avec la description des champs, il faut bien activer "Mes données ont des en-têtes".



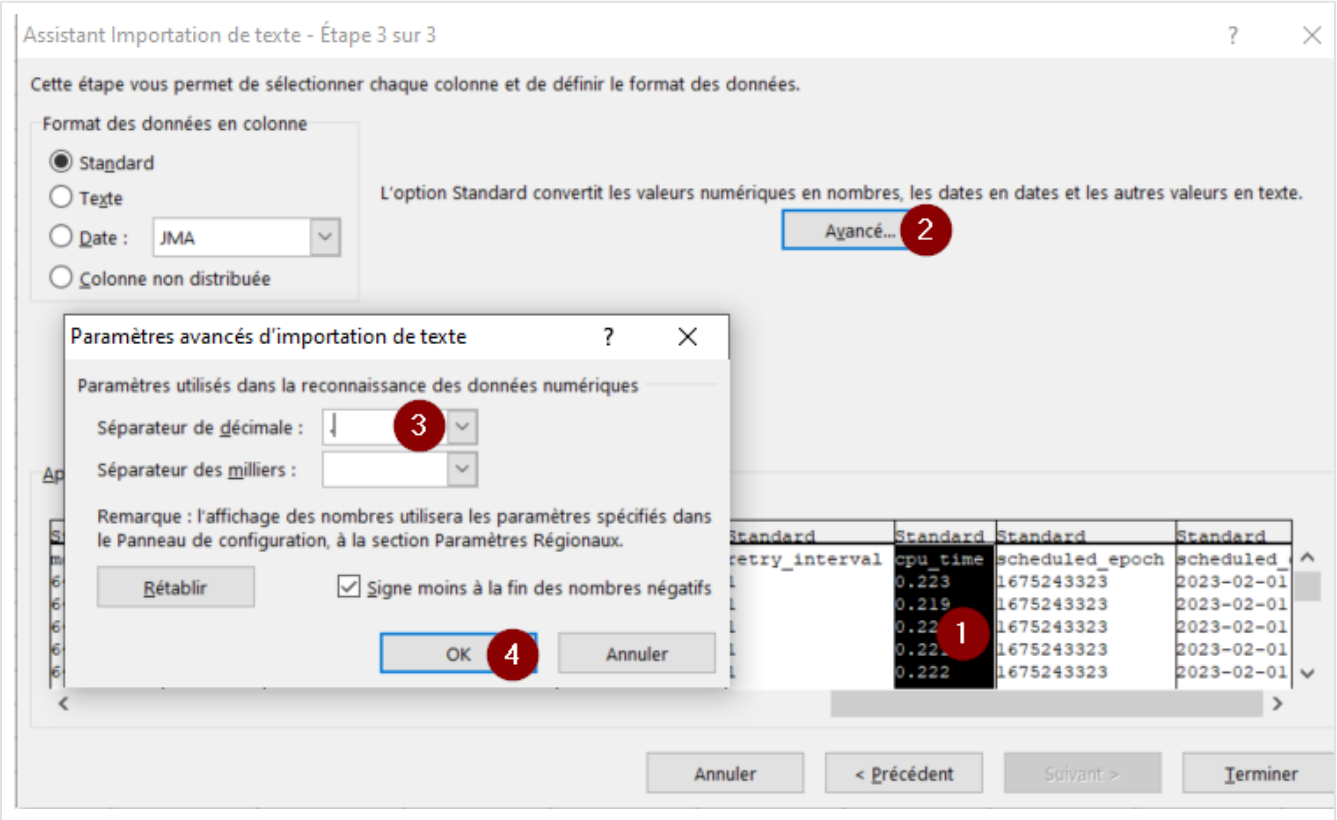
Sur la phase suivante, il faut bien placer le séparateur sur "Virgule".

- L'Aperçu de données permet de s'apercevoir que la séparation des champs est bien prise en compte.

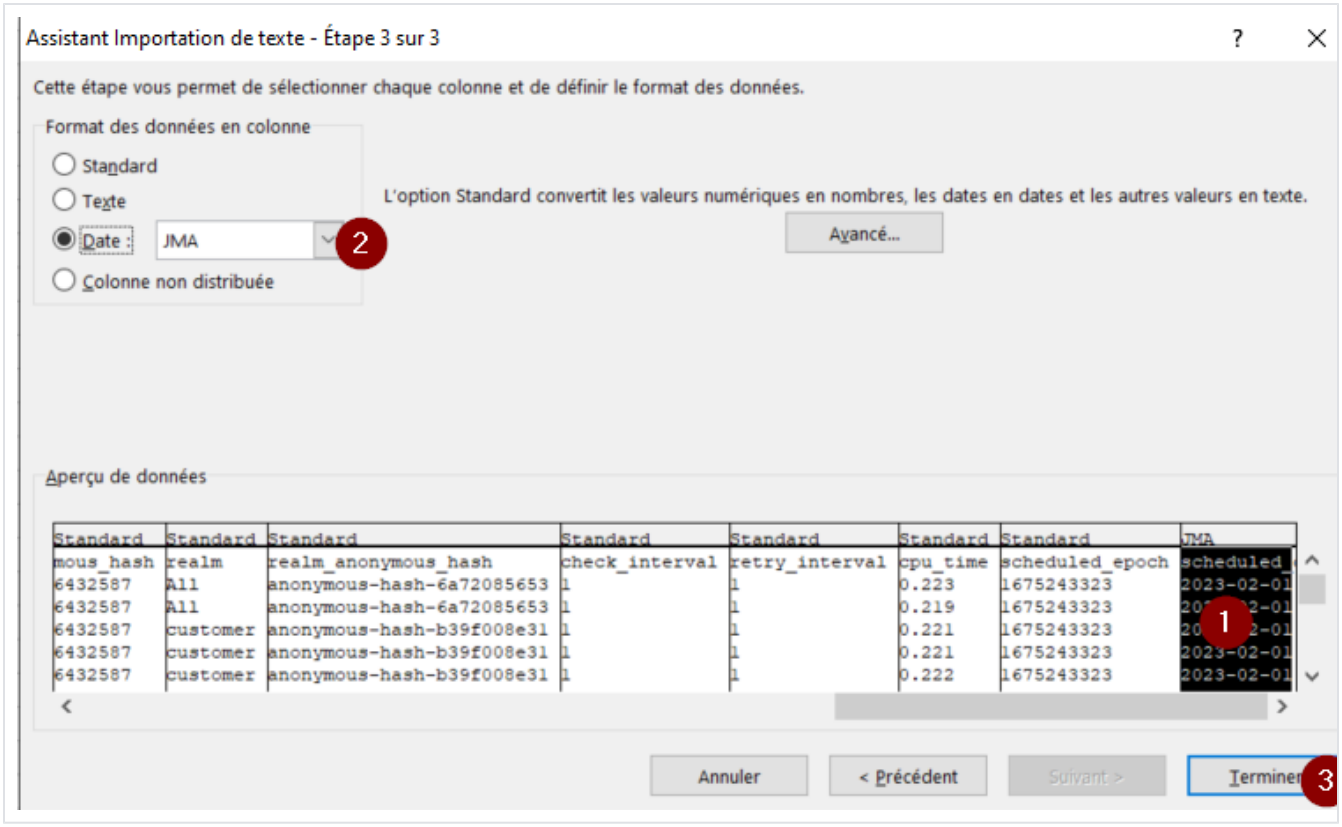


Si la majorité des champs sont des formats simples (*chaines de caractères ou entier*), deux colonnes sont particulières et on doit indiquer à Excel comment les interpréter :

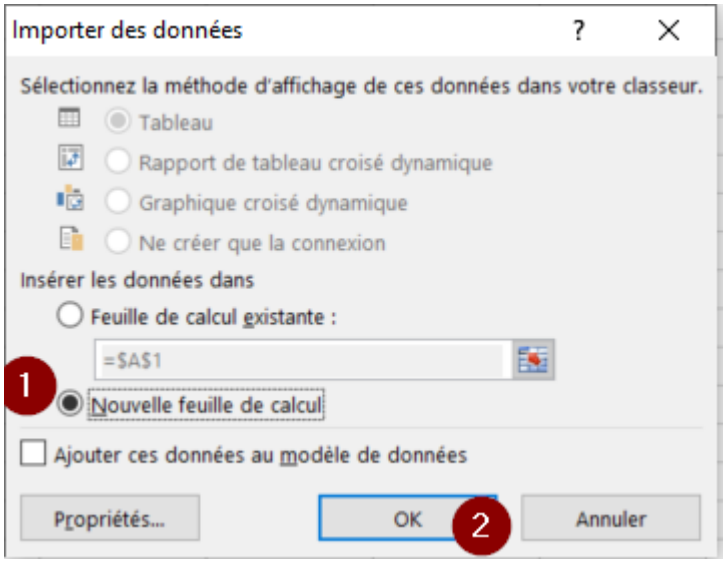
- **cpu_time** : Il faut lui indiquer via "Avancé" que le séparateur de décimale n'est pas **","** mais **"."** (*à faire que dans un Excel en Français, car dans le Excel en Anglais le séparateur par défaut est déjà le point*).



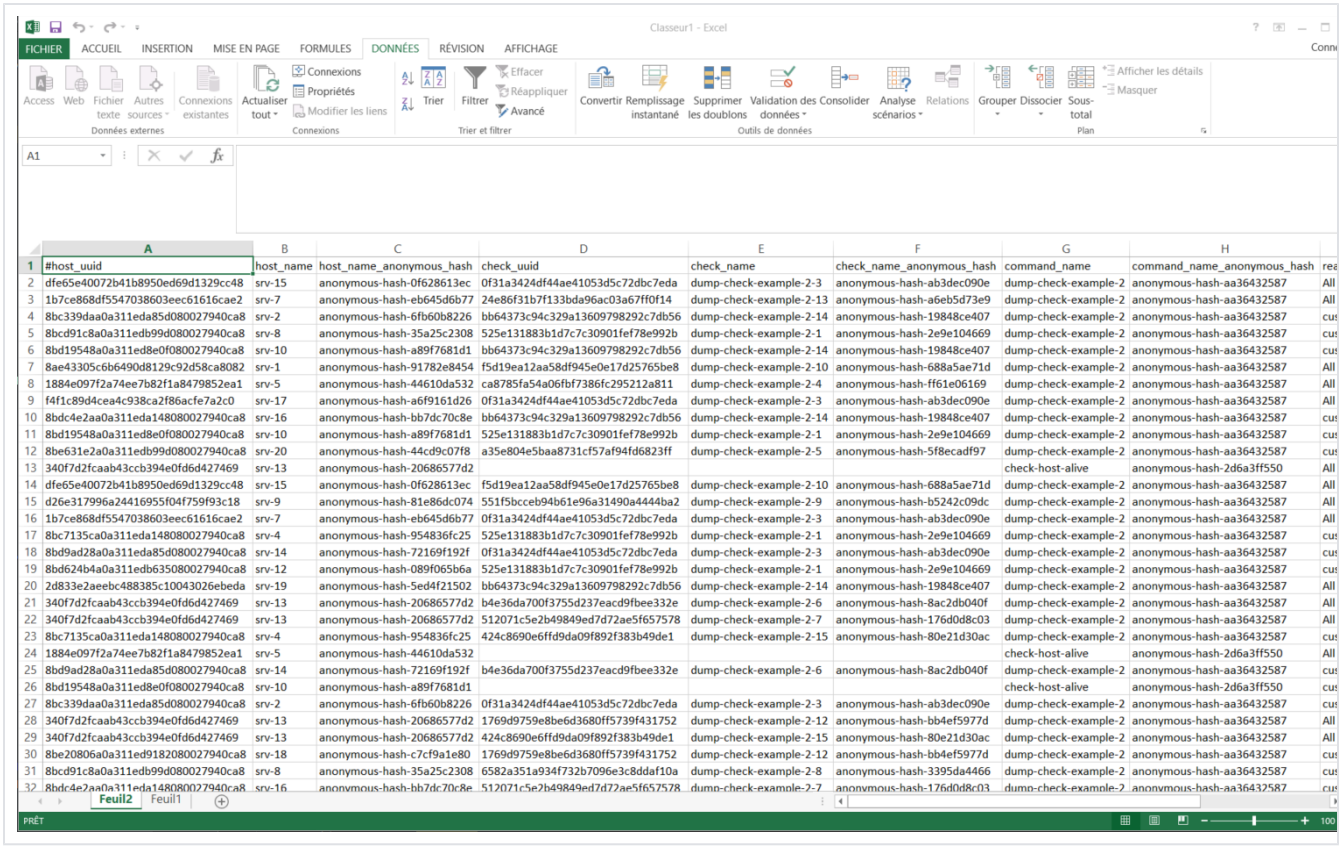
- **scheduled_date** : Il faut lui indiquer que le champ est un format Date.



Excel nous demande alors où charger ces données. Il est plus simple et lisible de le charger dans une "**Nouvelle feuille de calcul**" que l'on pourra renommer facilement.



Voici un exemple de ce qu'on obtient finalement :



#host_uuid	host_name	host_name_anonymous_hash	check_uuid	check_name	check_name_anonymous_hash	command_name	command_name_anonymous_hash	rea
dfe65e40072b41b8950ed69d1329cc48	srv-15	anonymous-hash-0f628613ec	0f31a3424df44ae41053d5c72dbc7eda	dump-check-example-2-3	anonymous-hash-ab3dec090e	dump-check-example-2	anonymous-hash-aa36432587	All
1b7ce868df5547038603eec61616cae2	srv-7	anonymous-hash-eb645d6b77	24e86f31b7f133bda96ac03a67ff0f14	dump-check-example-2-13	anonymous-hash-a6eb5d73e9	dump-check-example-2	anonymous-hash-aa36432587	All
8bc339daa0a311eda85d080027940ca8	srv-2	anonymous-hash-6fb60b8226	bb64373c94c329a13609798292c7db56	dump-check-example-2-14	anonymous-hash-19848ce407	dump-check-example-2	anonymous-hash-aa36432587	cus
8bcd91c8a0a311edb99d080027940ca8	srv-8	anonymous-hash-35a25c2308	525e131883b1d7c7c30901fef78e992b	dump-check-example-2-1	anonymous-hash-2e9e104669	dump-check-example-2	anonymous-hash-aa36432587	cus
8bd19548a0a311ed8e0f080027940ca8	srv-10	anonymous-hash-a89f7681d1	bb64373c94c329a13609798292c7db56	dump-check-example-2-14	anonymous-hash-19848ce407	dump-check-example-2	anonymous-hash-aa36432587	cus
8ae43305c6b6490d8129c92d58ca8082	srv-1	anonymous-hash-91782e8454	f5d19ea12aa58df945e0e17d25765be8	dump-check-example-2-10	anonymous-hash-688a5ae71d	dump-check-example-2	anonymous-hash-aa36432587	All
1884e097f2a74ee7b82f1a8479852ea1	srv-5	anonymous-hash-44610da532	ca8785fa54a06fbf7386fc295212a811	dump-check-example-2-4	anonymous-hash-ff61e06169	dump-check-example-2	anonymous-hash-aa36432587	All
f4f1c89d4cea4c938ca2f86acfe7a2c0	srv-17	anonymous-hash-a6f9161d26	0f31a3424df44ae41053d5c72dbc7eda	dump-check-example-2-3	anonymous-hash-ab3dec090e	dump-check-example-2	anonymous-hash-aa36432587	All
8bdc4e2aa0a311eda148080027940ca8	srv-16	anonymous-hash-bb7dc70c8e	bb64373c94c329a13609798292c7db56	dump-check-example-2-14	anonymous-hash-19848ce407	dump-check-example-2	anonymous-hash-aa36432587	cus
8bd19548a0a311ed8e0f080027940ca8	srv-10	anonymous-hash-a89f7681d1	525e131883b1d7c7c30901fef78e992b	dump-check-example-2-1	anonymous-hash-2e9e104669	dump-check-example-2	anonymous-hash-aa36432587	cus
8be631e2a0a311edb99d080027940ca8	srv-20	anonymous-hash-44cd9c07f8	a35e804e5baa8731cf57af94fd6823ff	dump-check-example-2-5	anonymous-hash-5f8ecadf97	dump-check-example-2	anonymous-hash-aa36432587	cus
340f7d2fcaab43ccb394e0fd6d427469	srv-13	anonymous-hash-20686577d2				check-host-alive	anonymous-hash-2d6a3ff550	All
dfe65e40072b41b8950ed69d1329cc48	srv-15	anonymous-hash-0f628613ec	f5d19ea12aa58df945e0e17d25765be8	dump-check-example-2-10	anonymous-hash-688a5ae71d	dump-check-example-2	anonymous-hash-aa36432587	All
d26e317996a24416955f04f759f93c18	srv-9	anonymous-hash-81e86dc074	551f5bcceb94b61e96a31490a4444ba2	dump-check-example-2-9	anonymous-hash-b5242c09dc	dump-check-example-2	anonymous-hash-aa36432587	All
1b7ce868df5547038603eec61616cae2	srv-7	anonymous-hash-eb645d6b77	0f31a3424df44ae41053d5c72dbc7eda	dump-check-example-2-3	anonymous-hash-ab3dec090e	dump-check-example-2	anonymous-hash-aa36432587	All
8bc7135ca0a311eda148080027940ca8	srv-4	anonymous-hash-954836fc25	525e131883b1d7c7c30901fef78e992b	dump-check-example-2-1	anonymous-hash-2e9e104669	dump-check-example-2	anonymous-hash-aa36432587	cus
8bd9ad28a0a311eda85d080027940ca8	srv-14	anonymous-hash-72169f192f	0f31a3424df44ae41053d5c72dbc7eda	dump-check-example-2-3	anonymous-hash-ab3dec090e	dump-check-example-2	anonymous-hash-aa36432587	cus
8bd624b4a0a311edb635080027940ca8	srv-12	anonymous-hash-089f065b6a	525e131883b1d7c7c30901fef78e992b	dump-check-example-2-1	anonymous-hash-2e9e104669	dump-check-example-2	anonymous-hash-aa36432587	cus
2d833e2aeebc488385c10043026ebeda	srv-19	anonymous-hash-5ed4f21502	bb64373c94c329a13609798292c7db56	dump-check-example-2-14	anonymous-hash-19848ce407	dump-check-example-2	anonymous-hash-aa36432587	All
340f7d2fcaab43ccb394e0fd6d427469	srv-13	anonymous-hash-20686577d2	b4e36da700f3755d237eacd9fbee332e	dump-check-example-2-6	anonymous-hash-8ac2db040f	dump-check-example-2	anonymous-hash-aa36432587	All
340f7d2fcaab43ccb394e0fd6d427469	srv-13	anonymous-hash-20686577d2	512071c5e2b49849ed7d72ae5f657578	dump-check-example-2-7	anonymous-hash-176d0d8c03	dump-check-example-2	anonymous-hash-aa36432587	All
8bc7135ca0a311eda148080027940ca8	srv-4	anonymous-hash-954836fc25	424c8690e6ffd9da09f892f383b49de1	dump-check-example-2-15	anonymous-hash-80e21d30ac	dump-check-example-2	anonymous-hash-aa36432587	cus
1884e097f2a74ee7b82f1a8479852ea1	srv-5	anonymous-hash-44610da532				check-host-alive	anonymous-hash-2d6a3ff550	All
8bd9ad28a0a311eda85d080027940ca8	srv-14	anonymous-hash-72169f192f	b4e36da700f3755d237eacd9fbee332e	dump-check-example-2-6	anonymous-hash-8ac2db040f	dump-check-example-2	anonymous-hash-aa36432587	cus
8bd19548a0a311ed8e0f080027940ca8	srv-10	anonymous-hash-a89f7681d1				check-host-alive	anonymous-hash-2d6a3ff550	cus
8bc339daa0a311eda85d080027940ca8	srv-2	anonymous-hash-6fb60b8226	0f31a3424df44ae41053d5c72dbc7eda	dump-check-example-2-3	anonymous-hash-ab3dec090e	dump-check-example-2	anonymous-hash-aa36432587	cus
340f7d2fcaab43ccb394e0fd6d427469	srv-13	anonymous-hash-20686577d2	1769d9759e8be6d3680ff5739f431752	dump-check-example-2-12	anonymous-hash-bb4ef5977d	dump-check-example-2	anonymous-hash-aa36432587	All
340f7d2fcaab43ccb394e0fd6d427469	srv-13	anonymous-hash-20686577d2	424c8690e6ffd9da09f892f383b49de1	dump-check-example-2-15	anonymous-hash-80e21d30ac	dump-check-example-2	anonymous-hash-aa36432587	All
8be20806a0a311ed9182080027940ca8	srv-18	anonymous-hash-c7cf9a1e80	1769d9759e8be6d3680ff5739f431752	dump-check-example-2-12	anonymous-hash-bb4ef5977d	dump-check-example-2	anonymous-hash-aa36432587	cus
8bcd91c8a0a311edb99d080027940ca8	srv-8	anonymous-hash-35a25c2308	6582a351a934f732b7096e3c8ddaf10a	dump-check-example-2-8	anonymous-hash-3395da4466	dump-check-example-2	anonymous-hash-aa36432587	cus
8bdc4e2aa0a311eda148080027940ca8	srv-16	anonymous-hash-bb7dc70c8e	512071c5e2b49849ed7d72ae5f657578	dump-check-example-2-7	anonymous-hash-176d0d8c03	dump-check-example-2	anonymous-hash-aa36432587	cus

Analyse des données

L'analyse des données issues du script va être répartie en deux grandes catégories :

- Vous avez un environnement avec un certain nombre d'hôtes, vous souhaitez dimensionner vos Pollers si vous rajoutez 10, 20 ou 50% d'hôtes en plus :
  - Dans ce cas, il faut se référer à la page Dimensionnement des Pollers à l'aide de la commande shinken-scheduler-export-data
- Si vous souhaitez diminuer le nombre de CPU de vos Pollers en identifiant, et optimisant, les sondes les plus consommatrices :
  - Dans ce cas, il faut se référer à la page Identification des checks les plus consommateurs à l'aide de la commande shinken-scheduler-export-data

Erreurs possibles

La commande est lancée depuis un serveur autre que l'Arbiter

La commande échange avec les Schedulers de la même manière que l'Arbiter. Elle a donc besoin d'avoir les accès réseaux. C'est pour cela que le lancement de la commande n'est autorisé que sur le serveur de l'Arbiter, car c'est le seul à avoir tous les accès vers les démons.

Dans le cas où la commande est lancée depuis un serveur qui n'est pas Arbiter, elle retourne l'erreur suivante :

```
ERROR: command must be launched on the Arbiter server
```

Problème de paramétrage du mot de passe d'accès (daemon__export_data__password dans /etc /shinken/shinken.cfg)

Par défaut, la commande va extraire les données avec les noms des éléments inclus (*désactivé avec la paramètre* ***--anonymous***). Vu que le nom des éléments est une information sensible à protéger, l'accès à ces informations est protégée par un système de mot de passe (*paramètre* ***daemon_ _export_data__password*** *dans* ***/etc/shinken/shinken.cfg***).

Le mot de passe est vide ou non configuré dans le fichier /etc/shinken/shinken.cfg

Dans le cas où le mot de passe est vide ou non paramétré (*le paramètre* ***daemon__export_data__password*** *dans le fichier* ***/etc/shinken/shinken.cfg***) vous aurez l'erreur suivante :

```
Cannot fetch SCHEDULER     : The "daemon__export_data__password" parameter is missing or void, that is
forbidden for not anonymous request
```

Le mot de passe fourni par l'utilisateur est incorrect

Dans le cas où le mot de passe rentré lors du lancement de la commande est incorrect, la commande retourne l'erreur suivante :

```
Cannot fetch SCHEDULER     : The password is not matching "daemon__export_data__password" parameter
```

Problème de paramétrage de la désactivation de l'export des données des démons (scheduler__export_data__enabled parameter dans le .cfg du Scheduler)

Il est possible de désactiver entièrement l'export de donnée sur un Scheduler (*depuis le paramètre **scheduler__export_data__enabled** dans le fichier de configuration du Scheduler*).

Dans le cas où on tente de communiquer avec un Scheduler avec cette configuration, la commande va retourner une erreur :

```
Cannot fetch SCHEDULER     : The Scheduler configuration disabled data export with the
"scheduler__export_data__enabled" parameter.
```

Problèmes avec le réseaux ou l'état du Scheduler

Le Scheduler n'est pas accessible (éteint ou problème réseau)

Dans le cas où le Scheduler est éteint ou s'il y a des problèmes réseaux entre le serveur de l'Arbiter et de dernier, l'erreur réseau de connexion sera retournée :

```
Cannot fetch SCHEDULER : ERREUR RESEAU ( connexion refusée, nom d'hôte inconnu, temps écoulé, ... )
```

Le Scheduler n'a pas encore reçu de configuration de l'Arbiter

Si le Scheduler n'a pas encore reçu de configuration de l'Arbiter:

- il n'a aucun éléments supervisés
- et il n'a pas reçu le mot de passe de vérification.

Dans le cas où la commande tente de se connecter au Scheduler avant qu'il ait reçu une communication, l'erreur suivante sera retournée :

```
Cannot fetch SCHEDULER     : The Scheduler does not have any configuration
```

Passer les erreurs et travailler avec des données partielles

Toutes les erreurs concernant les Schedulers sont par défaut bloquantes pour la commande afin d'être sûr d'avoir l'ensemble des données. Si un seul des Schedulers a une erreur, alors la commande sort en erreur et ne génère pas de fichier d'export csv.

Il est cependant possible, via le paramètre **--dont-stop-on-scheduler-error**, de signaler à la commande qu'on accepte les données partielles et que les erreurs sur les Schedulers ne sont que des avertissements et pas des erreurs bloquantes.

Dans le cas par défaut où on ne mets pas le paramètre, la commande indique qu'il est possible de les passer avec le message suivant :

```
There was some errors, cannot continue. Use --dont-stop-on-scheduler-error to allow partial results
```

Si on utilise le paramètre et qu'il y a eu des erreurs qu'on considère comme non bloquantes, alors :

- Les erreurs seront transformées en WARNING au niveau des Schedulers.
- La commande averti tout de même que les données risques d'être partielles avec le message suivant :

```
WARNING: The results may be partial because some schedulers were in errors:  SCHEDULERS
```