

# V2 - ( READ ) /api/v2/inventory

## Sommaire

- Objectifs
- Les Paramètres
- Réponse
  - Codes de retour
    - Retour du code 200
    - Retour du code 400
      - Paramètres POST incorrects
        - Paramètre inconnu
        - Paramètre désactivé sur cette route
    - Messages d'erreurs des filtres ( filterX )
      - Filtre inexistant
      - Filtre incomplet
      - Filtre incorrect
        - Valeur incorrecte pour ce type de filtre
        - Opérateur incorrect
        - Argument incorrect
    - Messages d'erreurs liés aux paramètres de tri ( sort )
      - Propriété inconnu
      - Ordre de tri incorrect
    - Messages d'erreurs liés au paramètre de format du résultat ( output\_format )
      - Valeur incorrecte
  - Erreurs communes lors de l'envoi de la requête
    - Messages d'erreurs liés au protocole HTTPS
      - Le certificat SSL a été refusé
      - Requête HTTP sur un serveur en HTTPS
      - Requête HTTPS sur un serveur en HTTP

## Objectifs

/api/v2/inventory est une Méthode POST de type READ qui permet de récupérer des données de supervision, comme sur l'Interface de Visualisation,

- Filtrées ( optionnel )
- Triées
- Rangées :
  - En arbres ( hôtes/clusters => checks )
  - Tous au même niveau

Le BUT de cette route étant d'obtenir un annuaire synthétique des éléments supervisés.

- Il n'est pas possible de choisir les propriétés affichées en sortie.

## Les Paramètres

Cet appel utilise seulement les 3 paramètres suivants :

- **filterX**
- **sort**
- **output\_format**

( voir la page [V2 - Les paramètres des API du broker-module-livedata](#) )

## Réponse

### Codes de retour

Codes de retour	Explications
200	OK
400	Paramètre invalide
401	Accès nécessite une authentification ou un Token valide.
403	Authentification de l'utilisateur OK , mais droits non suffisant.

## Retour du code 200

Les propriétés retournées sont :

- **nb\_element**
- **type** (si le paramètre **output\_format** vaut *elements\_on\_same\_level*)
- **father\_uuid**
- **father\_name**
- par check :
  - **check\_uuid**
  - **check\_name**

( voir la page [V2 - Les propriétés présentes dans le retour 200 des API du broker-module-livedata](#) )

Le retour aura une forme différente en fonction du paramètre **output\_format** :

### checks attached to father:

- **request\_statistics**
  - ...
- **elements\_found** :
  - **clusters** :
    - **cluster1**:
      - **father\_uuid** : *text*
      - **father\_name** : *text*
      - **checks** :
        - **check\_uuid**: *text*, **check\_name**: *text*
        - **check\_uuid**: *text*, **check\_name**: *text*
        - ...
    - ...
  - **hosts** :
    - **host1**:
      - **father\_uuid** : *text*
      - **father\_name** : *text*
      - **checks** :
        - **check\_uuid**: *text*, **check\_name**: *text*
        - **check\_uuid**: *text*, **check\_name**: *text*
        - ...
    - ...

### elements on same level:

- **request\_statistics**
  - ...
- **elements\_found** :
  - **elem1**:
    - **type**: *cluster*
    - **father\_name**: *text*
    - **father\_uuid** : *text*
  - **elem2**:
    - **type**: *host*
    - **father\_name**: *text*
    - **father\_uuid** : *text*
  - **elem3**:
    - **type**: *check\_cluster*
    - **check\_uuid** : *text*
    - **check\_name**: *text*
    - **father\_name** : *text*
    - **father\_uuid** : *text*
  - **elem4**:
    - **type**: *check\_host*
    - **check\_uuid** : *text*
    - **check\_name**: *text*
    - **father\_name** : *text*
    - **father\_uuid** : *text*
  - ...

```
curl -s -S -H 'x-api-token: XYZ' \
-d "output_format=checks_attached_to_father" \
http://broker-module-livedata:50100/api/v2
/inventory
```

```
curl -s -S -H 'x-api-token: XYZ' \
-d "output_format=elements_on_same_level" \
http://broker-module-livedata:50100/api/v2
/inventory
```

Exemple de de sortie attendue :

<http://localhost:50100/api/v2/inventory>

```
{
  "request_statistics": {
    "nb_elements_total": 9,
    "nb_hosts_total": 4,
    "nb_clusters_total": 1,
    "nb_checks_total": 4,
    "nb_elements_filtered": 9,
    "nb_hosts_filtered": 4,
    "nb_clusters_filtered": 1,
    "nb_checks_filtered": 4
  }
```

Exemple de de sortie attendue :

<http://localhost:50100/api/v2/inventory>

```
{
  "request_statistics": {
    "nb_elements_total": 9,
    "nb_hosts_total": 4,
    "nb_clusters_total": 1,
    "nb_checks_total": 4,
    "nb_elements_filtered": 9,
    "nb_hosts_filtered": 4,
    "nb_clusters_filtered": 1,
    "nb_checks_filtered": 4
  }
```

```

},
"elements_found": {
  "clusters": [
    {
      "father_uuid":
"d6921ee8ba1511eba36c0800277faebe",
      "father_name": "datacenter bdx",
      "checks": [
        {
          "check_name": "System Uptime",
          "check_uuid":
"d6921ee8ba1511eba36c0800277faebe-
e6daad4cba1511eb95980800277faebe"
        }
      ]
    }
  ],
  "hosts": [
    {
      "father_uuid":
"2c6dcflaba1611ebaa7d0800277faebe",
      "father_name": "Bordeaux",
      "checks": [
        {
          "check_name": "CPU Stats",
          "check_uuid":
"2c6dcflaba1611ebaa7d0800277faebe-
c296d75e5ad911e58cc5080027f08538"
        },
        {
          "check_name": "Disks Stats",
          "check_uuid":
"2c6dcflaba1611ebaa7d0800277faebe-
c29735965ad911e58cc5080027f08538"
        }
      ]
    },
    {
      "father_uuid":
"8db8b5f0ba1611eba60d0800277faebe",
      "father_name": "Lyon",
      "checks": []
    },
    {
      "father_uuid":
"76f45d80bale11eba2670800277faebe",
      "father_name": "Nantes",
      "checks": [
        {
          "check_name": "CPU Stats",
          "check_uuid":
"76f45d80bale11eba2670800277faebe-
c296d75e5ad911e58cc5080027f08538"
        }
      ]
    },
    {
      "father_uuid":
"e5460dc2ba1611eb81580800277faebe",
      "father_name": "Rennes",
      "checks": []
    }
  ]
}

```

```

},
"elements_found": [
  {
    "father_uuid":
"2c6dcflaba1611ebaa7d0800277faebe",
    "father_name": "Bordeaux",
    "type": "host"
  },
  {
    "check_name": "CPU Stats",
    "type": "check_host",
    "father_uuid":
"2c6dcflaba1611ebaa7d0800277faebe",
    "father_name": "Bordeaux",
    "check_uuid":
"2c6dcflaba1611ebaa7d0800277faebe-
c296d75e5ad911e58cc5080027f08538"
  },
  {
    "check_name": "Disks Stats",
    "type": "check_host",
    "father_uuid":
"2c6dcflaba1611ebaa7d0800277faebe",
    "father_name": "Bordeaux",
    "check_uuid":
"2c6dcflaba1611ebaa7d0800277faebe-
c29735965ad911e58cc5080027f08538"
  },
  {
    "father_uuid":
"d6921ee8ba1511eba36c0800277faebe",
    "father_name": "datacenter bdx",
    "type": "cluster"
  },
  {
    "check_name": "System Uptime",
    "type": "check_cluster",
    "father_uuid":
"d6921ee8ba1511eba36c0800277faebe",
    "father_name": "datacenter bdx",
    "check_uuid":
"d6921ee8ba1511eba36c0800277faebe-
e6daad4cba1511eb95980800277faebe"
  },
  {
    "father_uuid":
"8db8b5f0ba1611eba60d0800277faebe",
    "father_name": "Lyon",
    "type": "host"
  },
  {
    "father_uuid":
"76f45d80bale11eba2670800277faebe",
    "father_name": "Nantes",
    "type": "host"
  },
  {
    "check_name": "CPU Stats",
    "type": "check_host",
    "father_uuid":
"76f45d80bale11eba2670800277faebe",
    "father_name": "Nantes",
    "check_uuid":
"76f45d80bale11eba2670800277faebe-
c296d75e5ad911e58cc5080027f08538"
  },
  {
    "father_uuid":
"e5460dc2ba1611eb81580800277faebe",
    "father_name": "Rennes",
    "type": "host"
  }
]

```

```
}  
]  
}
```

## Retour du code 400

### Paramètres POST incorrects

#### Paramètre inconnu

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "parametre_inconnu=is_status:true" \  
http://broker-module-livadata:50100/api/v2  
/inventory  
ERROR 400: POST parameter [ parametre_inconnu ] is  
unknown
```

```
ERROR 400: POST parameter [ parametre_inconnu ] is unknown
```

#### Paramètre désactivé sur cette route

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "output_field=perf_data" \  
http://broker-module-livadata:50100/api/v2  
/inventory  
ERROR 400: POST parameter [ output_field ] is not  
available for this route
```

```
ERROR 400: POST parameter [ output_field ] is not available for this  
route
```

### Messages d'erreurs des filtres ( filterX )

( voir la page [V2 - Les paramètres des API du broker-module-livadata](#) )

#### Filtre inexistant

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "filter0=is_status:true" \  
http://broker-module-livadata:50100/api/v2  
/inventory  
ERROR 400: filtering[0]: invalid field name [  
is_status_ ]
```

```
ERROR 400: filtering[0]: invalid field name [ is_status_ ]
```

#### Filtre incomplet

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "filter0=next_check" \  
http://broker-module-livadata:50100/api/v2  
/inventory  
ERROR 400: filtering[0]: missing value for field [  
next_check ]
```

```
ERROR 400: filtering[0]: missing value for field [ next_check ]
```

#### Filtre incorrect

##### Valeur incorrecte pour ce type de filtre

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "filter0=status:9" \  
http://broker-module-livadata:50100/api/v2  
/inventory  
ERROR 400: filtering[0]: field [ status ] => wrong  
value ['9']
```

```
ERROR 400: filtering[0]: field [ status ] => wrong value ['9']
```

#### Opérateur incorrect

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "filter0=next_check:avant" \  
http://broker-module-livedata:50100/api/v2  
/inventory  
ERROR 400: filtering[0]: field [ next_check ]  
unknown date constraint [ avant ]
```

```
ERROR 400: filtering[0]: field [ next_check ] unknown date constraint  
[ avant ]
```

Argument incorrect

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "filter0=next_check:in-less-than|hier" \  
http://broker-module-livedata:50100/api/v2  
/inventory  
ERROR 400: filtering[0]: field [ next_check ] => [  
in-less-than ] => invalid literal for int() with  
base 10: 'hier'
```

```
ERROR 400: filtering[0]: field [ next_check ] => [ in-less-than ] =>  
invalid literal for int() with base 10: 'hier'
```

## Messages d'erreurs liés aux paramètres de tri ( sort )

( voir la page [V2 - Les paramètres des API du broker-module-livedata](#) )

Propriété inconnu

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "sort=host" \  
http://broker-module-livedata:50100/api/v2  
/inventory  
ERROR 400: sort: invalid field name [ host ]
```

```
ERROR 400: sort: invalid field name [ host ]
```

Ordre de tri incorrect

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "sort=father_name:big" \  
http://broker-module-livedata:50100/api/v2  
/inventory  
ERROR 400: sort: invalid sort direction [ big ]  
for field [ host_name ]
```

```
ERROR 400: sort: invalid sort direction [ big ] for field [ host_name ]
```

## Messages d'erreurs liés au paramètre de format du résultat ( output\_format )

( voir la page [V2 - Les paramètres des API du broker-module-livedata](#) )

Valeur incorrecte

```
$ curl -s -S -H "x-api-token: XYZ" \  
-d "output_format=vrai" \  
http://broker-module-livedata:50100/api/v2  
/inventory  
ERROR 400: output_format: invalid value [ vrai ]
```

```
ERROR 400: output_format: invalid value [ vrai ]
```

## Erreurs communes lors de l'envoi de la requête

### Messages d'erreurs liés au protocole HTTPS

Le certificat SSL a été refusé

```
$ curl -s -S -H "x-api-token: XYZ" \  
https://broker-module-livedata:50100/api/v2/inventory  
curl: (60) SSL certificate problem: unable to get  
local issuer certificate  
...
```



Cela peut se produire si le certificat du serveur est auto-signé.

Pour contourner la validité de l'émetteur du certificat, il faut utiliser l'option **--insecure** (*option courte : -k*).

```
curl --insecure -s -S -H "x-api-token: XYZ" \  
https://broker-module-livedata:50100/api/v2  
/inventory
```

Avec curl v7.29.0 :

```
curl: (60) Peer's Certificate issuer is not recognized.  
...
```

Avec curl v7.60 et supérieur :

```
curl: (60) SSL certificate problem: unable to get local issuer certificate  
...
```

### Requête HTTP sur un serveur en HTTPS

```
$ curl -s -S -H "x-api-token: XYZ" \  
http://broker-module-livedata:50100/api/v2/inventory  
The client IP_SERVEUR:PORT_CLIENT sent a plain HTTP  
request,  
but this server only speaks HTTPS on this port.
```

The client IP\_SERVEUR:PORT\_CLIENT sent a plain HTTP request,  
but this server only speaks HTTPS on this port.

### Requête HTTPS sur un serveur en HTTP

```
$ curl -s -S -H "x-api-token: XYZ" \  
https://broker-module-livedata:50100/api/v2/inventory  
curl: (35) error:1408F10B:SSL routines:  
ssl3_get_record:wrong version number
```

Avec curl v7.29.0 :

```
curl: (35) SSL received a record that exceeded the maximum  
permissible length.
```

Avec curl v7.60 et supérieur :

```
curl: (35) error:1408F10B:SSL routines:ssl3_get_record:wrong version  
number
```