

# item.get

## Description

integer/array `item.get(object parameters)`

La méthode permet de récupérer des éléments en fonction des paramètres donnés.

## Paramètres

(object) Paramètres définissant la sortie souhaitée.

La méthode prend en charge les paramètres suivants.

Paramètre	Type	Description
itemids	string/array	Renvoie uniquement les éléments avec les identifiants donnés.
groupids	string/array	Renvoie uniquement les éléments appartenant aux hôtes des groupes donnés.
templateids	string/array	Renvoie uniquement les éléments appartenant aux modèles donnés.
hostids	string/array	Renvoie uniquement les éléments appartenant aux hôtes donnés.
proxyids	string/array	Renvoie uniquement les éléments surveillés par les proxys donnés.
interfaceids	string/array	Renvoie uniquement les éléments utilisant les interfaces hôtes données.
graphids	string/array	Renvoie uniquement les éléments utilisés dans les graphiques donnés.
triggerids	string/array	Renvoie uniquement les éléments utilisés dans les déclencheurs donnés.
applicationids	string/array	Renvoie uniquement les éléments appartenant aux applications données.
webitems	flag	Inclure des éléments Web dans le résultat.
inherited	boolean	Si défini à <code>true</code> , ne renvoie que les éléments hérités d'un modèle.
templated	boolean	Si défini à <code>true</code> , ne renvoie que les éléments appartenant aux modèles.
monitored	boolean	Si défini à <code>true</code> , ne renvoie que les éléments activés appartenant aux hôtes surveillés.
group	string	Renvoie uniquement les éléments appartenant à un groupe portant le nom indiqué.
host	string	Renvoie uniquement les éléments appartenant à un hôte portant le nom indiqué.
application	string	Renvoie uniquement les éléments appartenant à une application portant le nom indiqué.
with_triggers	boolean	Si défini à <code>true</code> , ne renvoie que les éléments utilisés dans les déclencheurs.

Paramètre	Type	Description
selectHosts	query	Renvoie l'hôte auquel l'élément appartient en tant que tableau dans la propriété <code>hosts</code> .
selectInterfaces	query	Renvoie l'interface hôte utilisée par l'élément sous forme de tableau dans la propriété <code>interfaces</code> .
selectTriggers	query	Renvoie les déclencheurs qui utilisent cet élément dans la propriété <code>triggers</code> .  Supporte <code>count</code> .
selectGraphs	query	Renvoie les graphiques contenant l'élément dans la propriété <code>graphs</code> .  Supporte <code>count</code> .
selectApplications	query	Renvoie les applications auxquelles l'élément appartient dans la propriété <code>applications</code> .
selectDiscoveryRule	query	Renvoie la règle de découverte bas niveau qui a créé l'élément dans la propriété <code>discoveryRule</code> .
selectItemDiscovery	query	Renvoie l'objet de découverte d'élément dans la propriété <code>itemDiscovery</code> . L'objet de découverte d'élément lie l'objet à un prototype d'élément à partir duquel il a été créé.  Il a les propriétés suivantes : <code>itemdiscoveryid</code> - (string) Id de la découverte d'élément ; <code>itemid</code> - (string) ID de l'élément découvert ; <code>parent_itemid</code> - (string) ID du prototype d'élément à partir duquel l'élément a été créé ; <code>key_</code> - (string) clé du prototype d'élément ; <code>lastcheck</code> - (timestamp) moment où l'élément a été découvert pour la dernière fois ; <code>ts_delete</code> - (timestamp) moment où un élément qui n'est plus découvert sera supprimé.
selectPreprocessing	query	Renvoie les options de prétraitement des éléments dans la propriété <code>preprocessing</code> .  Il a les propriétés suivantes : <code>type</code> - (string) Les types d'option de prétraitement : 1 - Multiplicateur personnalisé ; 2 - Suppression des espaces à droite ; 3 - Suppression des espaces à gauche ; 4 - Suppression des espaces à droite et à gauche ; 5 - Correspondance d'expression régulière ; 6 - Booléen vers décimal ; 7 - Octal vers décimal ; 8 - Hexadécimal vers décimal ; 9 - Changement simple ; 10 - Changement par seconde.  <code>params</code> - (string) Paramètres supplémentaires utilisés par l'option de prétraitement. Plusieurs paramètres sont séparés par un caractère LF (\n).

Paramètre	Type	Description
filter	object	Renvoie uniquement les résultats qui correspondent exactement au filtre donné.  Accepte un tableau dans lequel les clés sont des noms de propriétés et les valeurs sont une valeur unique ou un tableau de valeurs à comparer.  Prend en charge des filtres supplémentaires : host - Nom technique de l'hôte auquel l'élément appartient.
limitSelects	integer	Limite le nombre d'enregistrements renvoyés par les sous-requêtes.  S'applique aux sous-requêtes suivantes : selectGraphs - les résultats seront triés par name ; selectTriggers - les résultats seront triés par description.
sortfield	string/array	Trie le résultat par les propriétés données.  Les valeurs possibles sont : itemid, name, key_, delay, history, trends, type et status.
countOutput	boolean	Ces paramètres, communs à toutes les méthodes "get", sont décrits dans les <a href="#">commentaires de référence</a> .
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

## Valeurs retournées

(integer/array) Retourne soit :

- un tableau d'objets ;
- le nombre d'objets récupérés, si le paramètre countOutput a été utilisé.

## Exemples

### Recherche d'éléments par clé

Récupérer tous les éléments de l'hôte avec l'ID "10084" contenant le mot "system" dans la clé et les trier par nom.

Requête :

```
{
  "jsonrpc": "2.0",
  "method": "item.get",
  "params": {
    "output": "extend",
    "hostids": "10084",
    "search": {
      "key_": "system"
    },
    "sortfield": "name"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Réponse :

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "23298",
      "type": "0",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10084",
      "name": "Context switches per second",
      "key_": "system.cpu.switches",
      "delay": "1m",
      "history": "7d",
      "trends": "365d",
      "lastvalue": "2552",
      "lastclock": "1351090998",
      "prevvalue": "2641",
      "state": "0",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "sps",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "snmpv3_contextname": "",
      "error": "",
      "lastlogsize": "0",
      "logtimefmt": ""
    }
  ]
}
```

```
    "templateid": "22680",
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "mtime": "0",
    "lastns": "564054253",
    "flags": "0",
    "interfaceid": "1",
    "port": "",
    "description": "",
    "inventory_link": "0",
    "lifetime": "0s",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "timeout": "3s",
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0"
  },
  {
    "itemid": "23299",
    "type": "0",
    "snmp_community": "",
    "snmp_oid": "",
    "hostid": "10084",
    "name": "CPU $2 time",
    "key_": "system.cpu.util[,idle]",
    "delay": "1m",
    "history": "7d",
    "trends": "365d",
    "lastvalue": "86.031879",
```

```
"lastclock": "1351090999",
"prevvalue": "85.306944",
"state": "0",
"status": "0",
"value_type": "0",
"trapper_hosts": "",
"units": "%",
"snmpv3_securityname": "",
"snmpv3_securitylevel": "0",
"snmpv3_authpassphrase": "",
"snmpv3_privpassphrase": "",
"snmpv3_authprotocol": "0",
"snmpv3_privprotocol": "0",
"snmpv3_contextname": "",
"error": "",
"lastlogsize": "0",
"logtimefmt": "",
"templateid": "17354",
"valuemapid": "0",
"params": "",
"ipmi_sensor": "",
"authtype": "0",
"username": "",
"password": "",
"publickey": "",
"privatekey": "",
"mtime": "0",
"lastns": "564256864",
"flags": "0",
"interfaceid": "1",
"port": "",
"description": "The time the CPU has spent doing nothing.",
"inventory_link": "0",
"lifetime": "0s",
"evaltype": "0",
"jmx_endpoint": "",
"master_itemid": "0",
"timeout": "3s",
"url": "",
"query_fields": [],
"posts": "",
"status_codes": "200",
"follow_redirects": "1",
"post_type": "0",
"http_proxy": "",
"headers": [],
"retrieve_mode": "0",
"request_method": "0",
"output_format": "0",
"ssl_cert_file": "",
```

```
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0"
  },
  {
    "itemid": "23300",
    "type": "0",
    "snmp_community": "",
    "snmp_oid": "",
    "hostid": "10084",
    "name": "CPU $2 time",
    "key_": "system.cpu.util[,interrupt]",
    "history": "7d",
    "trends": "365d",
    "lastvalue": "0.008389",
    "lastclock": "1351091000",
    "prevvalue": "0.000000",
    "state": "0",
    "status": "0",
    "value_type": "0",
    "trapper_hosts": "",
    "units": "%",
    "snmpv3_securityname": "",
    "snmpv3_securitylevel": "0",
    "snmpv3_authpassphrase": "",
    "snmpv3_privpassphrase": "",
    "snmpv3_authprotocol": "0",
    "snmpv3_privprotocol": "0",
    "snmpv3_contextname": "",
    "error": "",
    "lastlogsize": "0",
    "logtimefmt": "",
    "templateid": "22671",
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "mtime": "0",
    "lastns": "564661387",
    "flags": "0",
    "interfaceid": "1",
    "port": "",
    "description": "The amount of time the CPU has been servicing hardware interrupts.",
    "inventory_link": "0",
```

```
    "lifetime": "0s",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "timeout": "3s",
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0"
  }
],
  "id": 1
}
```

## Recherche d'éléments dépendants par clé

Récupérer tous les éléments dépendants de l'hôte avec l'ID "10116" contenant le mot "apache" dans la clé.

Requête :

```
{
  "jsonrpc": "2.0",
  "method": "item.get",
  "params": {
    "output": "extend",
    "hostids": "10116",
    "search": {
      "key_": "apache"
    },
    "filter": {
      "type": "18"
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```



```
}
```

Réponse :

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "25550",
      "type": "18",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10116",
      "name": "Days",
      "key_": "apache.status.uptime.days",
      "delay": "",
      "history": "90d",
      "trends": "365d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "formula": "",
      "error": "",
      "lastlogsize": "0",
      "logtimefmt": "",
      "templateid": "0",
      "valuemapid": "0",
      "params": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "mtime": "0",
      "flags": "0",
      "interfaceid": "0",
      "port": "",
      "description": "",
      "inventory_link": "0",
      "lifetime": "30d",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "state": "0",
      "snmpv3_contextname": "",
    }
  ]
}
```

```
"evaltype": "0",
"master_itemid": "25545",
"jmx_endpoint": "",
"timeout": "3s",
"url": "",
"query_fields": [],
"posts": "",
"status_codes": "200",
"follow_redirects": "1",
"post_type": "0",
"http_proxy": "",
"headers": [],
"retrieve_mode": "0",
"request_method": "0",
"output_format": "0",
"ssl_cert_file": "",
"ssl_key_file": "",
"ssl_key_password": "",
"verify_peer": "0",
"verify_host": "0",
"allow_traps": "0",
"lastclock": "0",
"lastns": "0",
"lastvalue": "0",
"prevvalue": "0"
},
{
  "itemid": "25555",
  "type": "18",
  "snmp_community": "",
  "snmp_oid": "",
  "hostid": "10116",
  "name": "Hours",
  "key_": "apache.status.uptime.hours",
  "delay": "0",
  "history": "90d",
  "trends": "365d",
  "status": "0",
  "value_type": "3",
  "trapper_hosts": "",
  "units": "",
  "snmpv3_securityname": "",
  "snmpv3_securitylevel": "0",
  "snmpv3_authpassphrase": "",
  "snmpv3_privpassphrase": "",
  "formula": "",
  "error": "",
  "lastlogsize": "0",
  "logtimefmt": "",
  "templateid": "0",
```

```
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "mtime": "0",
    "flags": "0",
    "interfaceid": "0",
    "port": "",
    "description": "",
    "inventory_link": "0",
    "lifetime": "30d",
    "snmpv3_authprotocol": "0",
    "snmpv3_privprotocol": "0",
    "state": "0",
    "snmpv3_contextname": "",
    "evaltype": "0",
    "master_itemid": "25545",
    "jmx_endpoint": "",
    "timeout": "3s",
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "lastclock": "0",
    "lastns": "0",
    "lastvalue": "0",
    "prevvalue": "0"
  }
],
  "id": 1
}
```

## Recherche d'élément d'agent HTTP

Rechercher l'élément d'agent HTTP avec le type de corps de message XML pour un ID d'hôte spécifique.

Requête :

```
{
  "jsonrpc": "2.0",
  "method": "item.get",
  "params": {
    "hostids": "10255",
    "filter": {
      "type": "19",
      "post_type": "3"
    }
  },
  "id": 3,
  "auth": "d678e0b85688ce578ff061bd29a20d3b"
}
```

Réponse :

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "28252",
      "type": "19",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10255",
      "name": "template item",
      "key_": "ti",
      "delay": "30s",
      "history": "90d",
      "trends": "365d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "formula": "",
      "error": ""
    }
  ]
}
```

```
"lastlogsize": "0",
"logtimefmt": "",
"templateid": "0",
"valuemapid": "0",
"params": "",
"ipmi_sensor": "",
"authtype": "0",
"username": "",
"password": "",
"publickey": "",
"privatekey": "",
"mtime": "0",
"flags": "0",
"interfaceid": "0",
"port": "",
"description": "",
"inventory_link": "0",
"lifetime": "30d",
"snmpv3_authprotocol": "0",
"snmpv3_privprotocol": "0",
"state": "0",
"snmpv3_contextname": "",
"evaltype": "0",
"jmx_endpoint": "",
"master_itemid": "0",
"timeout": "3s",
"url": "localhost",
"query_fields": [
  {
    "mode": "xml"
  }
],
"posts": "<body>\r\n<![CDATA[{$MACRO}<foo></bar>]]>\r\n</body>",
"status_codes": "200",
"follow_redirects": "0",
"post_type": "3",
"http_proxy": "",
"headers": [],
"retrieve_mode": "1",
"request_method": "3",
"output_format": "0",
"ssl_cert_file": "",
"ssl_key_file": "",
"ssl_key_password": "",
"verify_peer": "0",
"verify_host": "0",
"allow_traps": "0",
"lastclock": "0",
"lastns": "0",
"lastvalue": "0",
"prevvalue": "0"
```

```
    },  
    "id": 3  
  }  
}
```

## Voir également

- [Application](#)
- [Règle de découverte bas niveau](#)
- [Graphique](#)
- [Hôte](#)
- [Interface hôte](#)
- [Déclencheur](#)

## Source

CItem::get() dans *frontends/php/include/classes/api/services/CItem.php*.

From:

<https://www.zabbix.com/documentation/4.0/> - **Zabbix Documentation 4.0**

Permanent link:

<https://www.zabbix.com/documentation/4.0/fr/manual/api/reference/item/get>

Last update: **2019/03/08 10:43**

